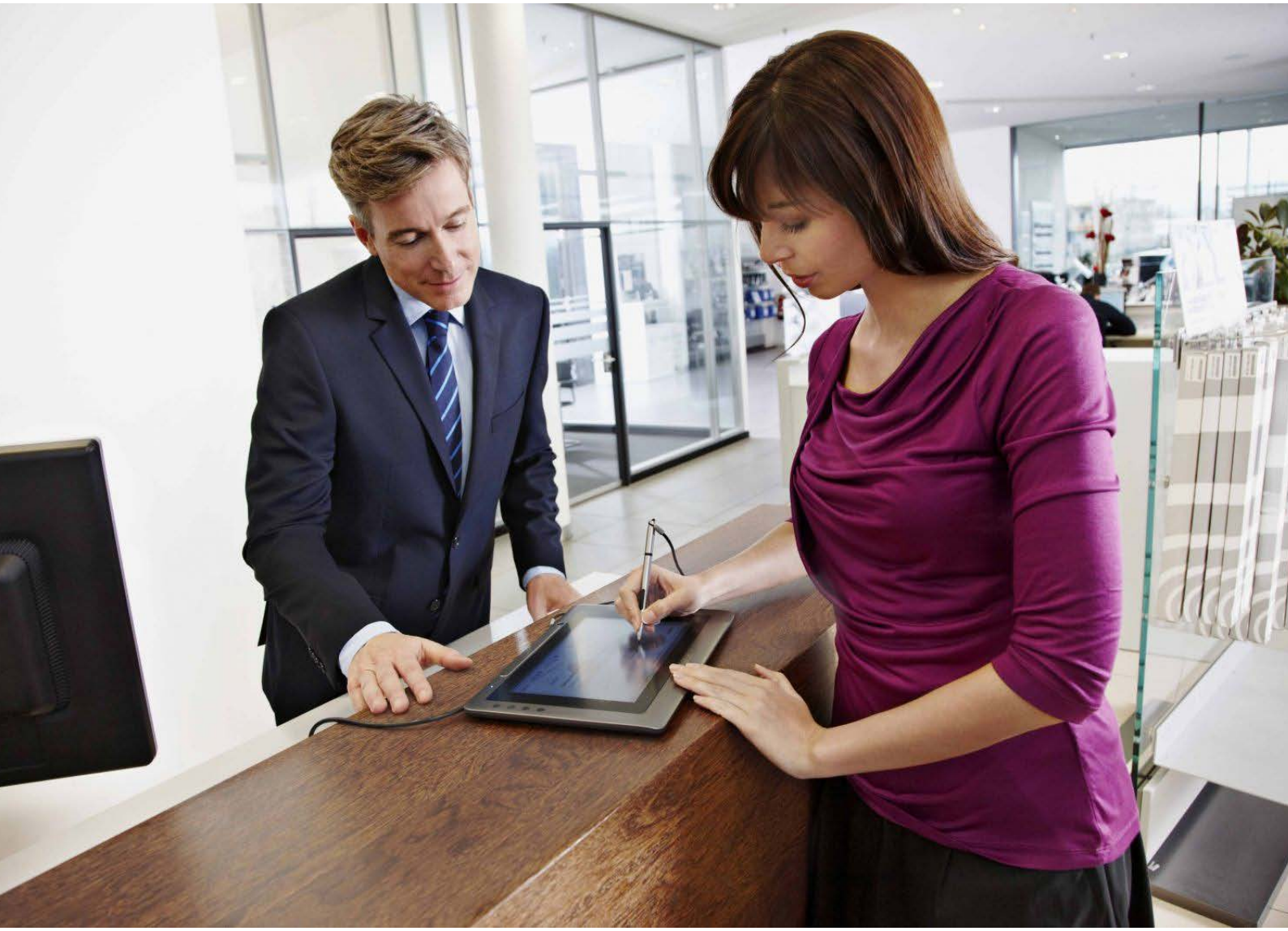
E-Signing at the Point of Sale

Paperless B2C Contracting through Direct or Indirect Sales Channels



NAMIRIAL GmbH

Legal Office: Seilerstätte 16, 1010 Wien, Austria
Main Office: Haider Straße 23, 4025 Ansfelden | Phone: +43-7229-88060 | www.xyzmo.com
Fiscalnumber 09 258/9720 | VAT-ID: ATU70125036

Abstract

In today's competitive business climate, it is essential to seek cost-cutting possibilities, to improve operational efficiency, and to pay attention to customer interests and demands to improve the bottom line. Printing documents just to capture a customer signature is not only completely outdated in today's tablet-pervasive everyday life but also is a great waste of time and money. More than that, paper handling is extremely time consuming for sales and service personnel and thus reduces the likelihood for efficient customer communication, which in turn limits up-sell and cross-sell opportunities.

Modern e-signature-based digital document processes are now geared up to remedy the situation, as they are able to close the final gap in the quest to go fully paperless at the point of sale (POS). This white paper looks at the specific requirements for such e-signature software in typical business-to-consumer (B2C) use cases in both direct sales channels as they are typically found in today's bank branches, retail stores, and customer centers and indirect sales channels, such as those operated by agencies and merchants.

First, this whitepaper helps you select the most appropriate way to e-sign your digital documents in your POS scenario. After pointing out why you need to look beyond pure e-signature capturing toward productivity, we highlight the most important security aspects you need to consider. After introducing the pros and cons of various hardware choices for signature capturing, we show you how to ensure the authenticity of signed documents and how to prove the validity of documents in case of a dispute. Next, we touch upon the most important topics you need to think about when integrating and deploying the e-signature solution into your IT and application environment. Finally, the paper introduces the SIGNificant e-signature platform and outlines a few case studies that show different implementations in stationary POS scenarios across the industry.

Table of Contents

1 Selecting the Right Methodology

Today, there are quite a few different e-signature solutions for direct and indirect point-of-sale (POS) processes available on the market. While they all allow users to sign documents digitally, their approaches can be differentiated in the following three key areas:

- e-signing technology
- document format
- software architecture

As a first step, we want to examine the different available options for each of these categories.

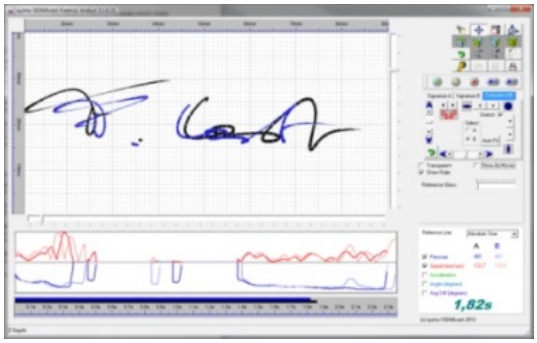
1.1 E-signing technology

The most popular e-signature technologies for B2C processes at the POS are:

- **Forensically identifiable signatures** (aka biometric signatures) in which the unique characteristics of real handwritten signatures are captured (e.g., speed, acceleration, pressure) that allow for a signature verification by a graphologist. The process of signing and signature verification is basically the same as it is with paper-based signatures.
- **HTML5 signatures** in which signers also sign with their handwritten signatures as they do on paper. However, e-signing software—due to HTML5 limitations—cannot record reliable forensic data, which reduces the traceability of the signature's image, making an additional user authentication (e.g., one-time password, SMS-TAN, ID verification, etc.) inevitable, which must be included together with the signature in some kind of audit trail. Thus, both the process of signing and the process of signature verification are quite different from today's paper-based processes.
- **Certificate-based signatures** that require a public key infrastructure (PKI) that provides personal digital signing certificates to potential users (e.g., using smart cards or online access). The signing process here is entirely different from that of handwritten signatures (on paper) and is more comparable to passport authentication at a border or entrance control.

In a B2C POS scenario, PKI-based approaches do not work well. One reason might be that the penetration even of "national ID-cards" – including signature function - is still quite low, most likely because of the costs and inconvenience of such approaches, especially to people who are not used to working with the latest technologies. Consequently, one must expect that potential clients either do not own a personal signing certificate or cannot/don't want use it (e.g., because they forgot the access PIN or the smart card that stores the certificate).

HTML5 signatures, in contrast, are best suited for B2C processes in which the client needs to sign a document remotely without meeting a sales person face to face because these signatures do not require any upfront installation, meaning that clients can easily sign on their own devices (e.g., smartphone or PC). In this case, the extra step of authenticating via another method different from the pure signature-writing act (e.g., drawing/writing the name) seems to be acceptable to clients.



Capturing a forensically identifiable handwritten signature remains the best choice for getting documents signed in person, in a meeting with the customer. Although there are other biometric technologies available, biometric signature has finally emerged as the de facto industry standard for electronic signatures in B2C environments because handwritten signatures are socially widely accepted and capturing their biometrical data is seen as nonintrusive for the masses—especially when the signing environment at the POS is preinstalled and thus ready to use and the basic process for a consumer is the same as it is on paper, thus no need to adjust to something new.

1.2 Document format

According to Gartner Research (Publication ID Number: G00159721) the best document format is self-contained. Thus it includes the content to be signed, the signature, and the metadata to make it searchable, and it stores the information needed for proof in addition to the signature—which is date, time, and consent in its internal audit log. It should also only require a freely and ubiquitously available reader to show the document in its originally archived form.

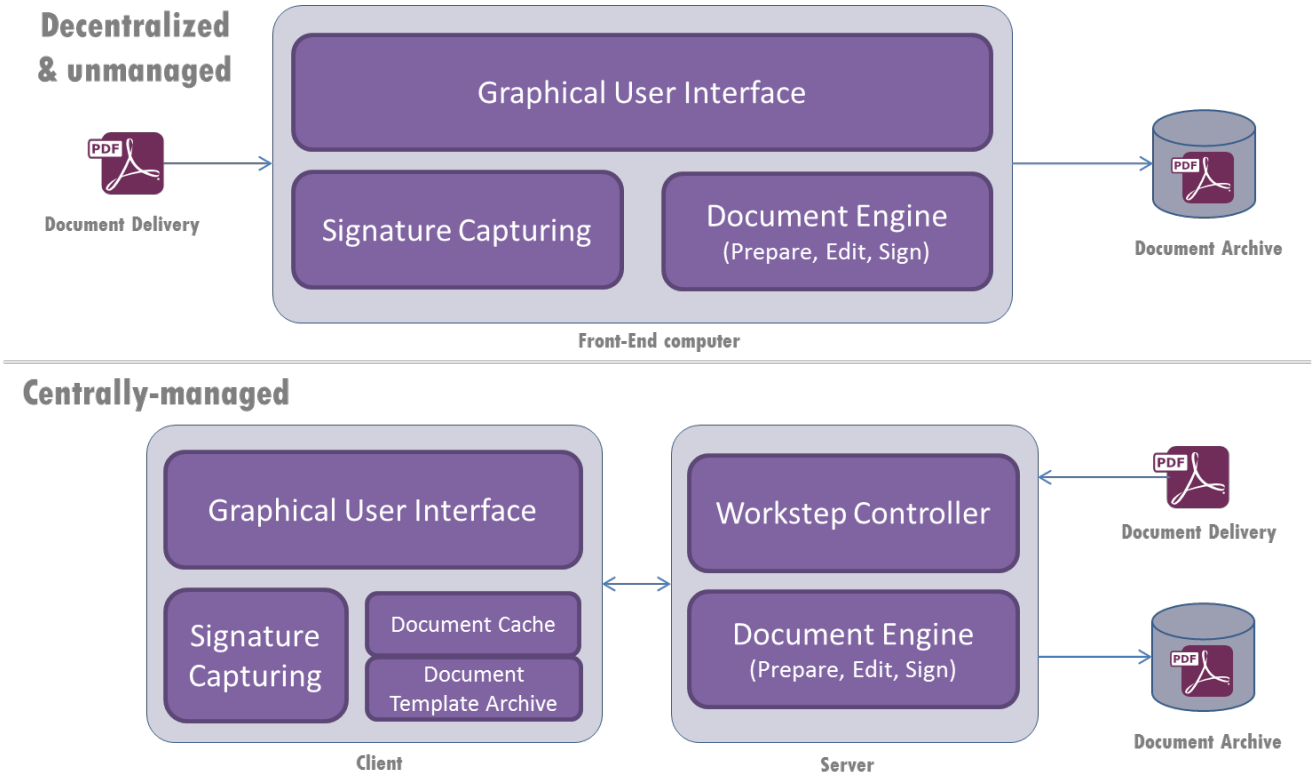
Other than proprietary document formats and document databases, the open portable document format (PDF) fulfills all these requirements. PDF is not only an open standard defined in ISO 32000-1:2008, but also it comes in a variant designed for long-time archiving defined as a PDF/A in ISO 19005-1:2005. Additionally, digital signatures are well defined within the PDF itself (Adobe PDF Reference PDF 32000-1:2008 12.8.3.3 PKCS#7 Signatures—as used in ISO 32000), meaning that every standard compliant viewing application such as Adobe Acrobat Reader correctly shows digitally signed PDFs. As such, a PDF or PFD/A file is the perfect pendant to paper in the digital world for archiving signed document originals.

1.3 Software architecture

An e-signing application typically consists of a front-end and a back-end component. While the front-end software manages all user interactions, the back-end software processes the document and takes care of its integration into the overall document workflow.

The front-end software component naturally runs on a POS computing device, which can be either a PC or a tablet computer. The back-end software component can run either

locally together with the front-end component inside the same application/on the same computer or be split off into a separate server application, which means that the e-singing application is distributed over a client and a server.



SIGNificant – Architectural Options

In many scenarios, the client/server model with a centralized back-end software component has many advantages over any e-singing software that is installed separately on each local POS computing device. These include:

- If existing systems for document creation, workflow management, and document archiving are also server-based, the server-side integration is simply much easier.
- The PDF document is only stored in the secure data center and not automatically distributed to the clients, where access to the signed original cannot be securely managed.
- A rich server-side audit trail providing additional process evidence
- A server provides a single point and type of integration for all the different client options:
    - signature pads—managed by a web application or local SDK to be integrated in custom-rich client application
    - signature screens—controlled by a local Kiosk SDK that you can also integrate easily into a your own Web application
    - smartphones—that run a small signature capture app that connects with a Web application to view the document
    - tablets—that run native signing clients to display, edit, and sign documents
- Compatibility to additional sales channels—thus reusing the e-singing infrastructure and software integrations already implemented for the POS in a multichannel environment that also includes mobile and online channels

In addition, many companies even centralize their front-end software through terminal service solutions, such as those from Citrix or Microsoft, because they make software deployment and management much easier.

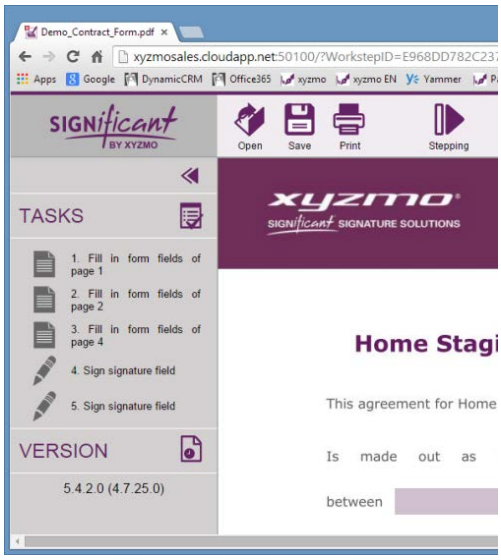
In contrast, purely desktop-/local-based signing approaches are typically preferred if:

- the document to be signed is dynamically created on the client, meaning that transferring it to the signing server, before processing it on the client, would introduce an additional step;
- server-side integration is not necessary at all;
- poor network connectivity to the clients, due to a low network bandwidth and high latency, is a big issue. However, this point can be widely ironed out, e.g., through local document templates, caching and background syncing.

2 More than just Capturing a Signature

Completing a contract sometimes not just involves signing but also potentially entails editing and filling out the document itself. The more complex this is the higher the chance is to create an ill-completed contract, which makes a proper guiding highly desirable to avoid a situation in which you discover forgotten signature or form fields after the client has already left.

2.1 Avoiding incomplete contracts



Trying to fix ill-signed contracts afterwards is often extremely time consuming and costly because when you discover the problem, the client typically is long gone and not easily accessible any more. Thus, it is a huge benefit if you can control and govern all necessary steps in the completion and signing process of documents, including filling out form fields, reading important clauses, accessing scanners or the camera for adding attachments such as ID scans, signing on signature fields, and much more. Ideally, you can specify compulsory or optional tasks depending on the use case and document, thus allowing you the flexibility you need to best cover all your business cases.

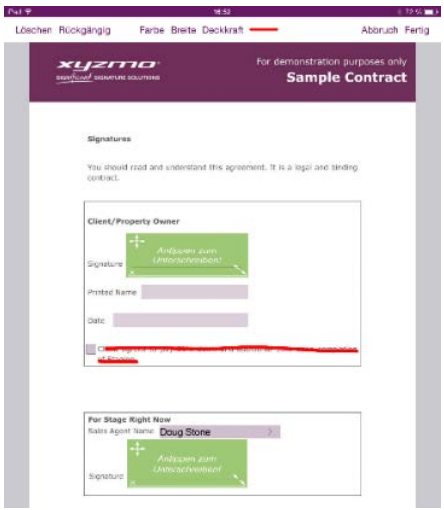
Additionally, through defining policies that enable or forbid certain actions on or with the document, such as making annotations, saving, e-mailing, or printing documents, you can exercise any required further control.

2.2 Form Fields and Attachments

Thus, it is important that the e-signature solution allows you to fill out form fields such as checkboxes or text fields and that you can add items such as scanned images or other files (either

as a visible item e.g., on a new page or as an attachment). The data and attachments then must be sealed into the document with every provided signature, which ensures that any subsequent change would be visible in the document. When completing those user actions with the e-signing software, you can further increase your process evidence (see chapter 5.3.1) by logging those user actions into an integrated audit trail.

2.3 Free document edits (annotations)



It is not always possible to include all information in documents in advance or to parameterize them via form fields. The e-signing solution must allow you to add annotations - such as specifications - either via a typewriter or with free hand.

Other examples are complex contracts that require a more consultative approach, in which you want to allow either party to highlight certain key areas (e.g., on a photo), make drawings in the document, or apply last minute changes to the document. Tools for annotations with free hand or typewriter simply provide you with the flexibility you need.

2.4 Allow document reading and editing as if on paper

Ideally, clients want to work with digital documents in the same way as they are used to working with paper documents. This means that the e-signing application certainly must allow clients to review multipage documents before signing them—ideally directly on the signing device.With mobile tablets (see chapter 4.2.3), you can easily go beyond this, as they also allow editing documents the way you are used to in the paper world. This includes free hand and text annotations, attachments, and filling out form fields.

Also, the integration of the tablet-based signing solution with the document workflow is key, as you may want to push a pre-filled form document (e.g., a client contract) from a POS PC to a specific tablet device, then allow the client to read and update its form field values, and sign it. After that, all updates the client made to the form field values are saved back into your own database.

3 Security Aspects

As the signed documents are from now on your legally bound originals. Security has to be bulletproof; otherwise, the digital originals become worthless. Therefore, security aspects are a major topic. The most important aspects are pointed out in this chapter. For more detailed information, please ask for the SIGNificant security whitepaper.

3.1 Authenticity protection

Protecting the authenticity of a signature and its binding to a certain document and position within a document is core to all security aspects of e-signing. It simply must not be possible for an attacker to access and copy the signature data of one document and paste it somewhere else—whether it be within the same document or into a new document. Thus, secure encryption of the raw data—the captured biometric signature—together with the document fingerprint (= hash value) is critical.

Here, asymmetrical encryption using a hybrid RSA/AES encryption algorithm is viewed generally safe and has been emerged as the de facto industry standard. Today, nearly all important signature capture devices (see chapter 4) can perform these asymmetric encryption operations directly on the devices themselves, thus efficiently preventing wiretapping of the biometric signature data.

Naturally, proving the document's signature binding should also not depend on the availability of the signature-capturing device on which the signature was captured, because signed documents have a much longer time span than the those devices do.

3.2 Integrity protection

Once a document is signed, it is essential that it can easily be determined whether the signed document is still original or whether it has been altered after the signature has been applied. This kind of integrity analysis must be easily available to everyone who is viewing/reading the signed document; otherwise, forging the content of signed documents is as easy as it is on paper.

3.3 Limiting access to documents

In contrast to paper, digital files can be easily copied without losing any of their characteristics. If a digital file is an original, a digital copy of it creates another valid original. In case you want to limit access to an original signed document for security reasons, you must make sure that the e-signing solution does not simply distribute the original file to all decentralized signing stations—which would significantly increase the complexity of securing access to the signed original.

3.4 Option to verify a signature in real time for the highest process security

In addition to the deep manual signature verification a graphologist can conduct in case of a legal dispute at any time after the document was signed, you can also authenticate a signer in real time – straight after the signing process - and document it in a secure audit log (see chapter 5). With this real-time signature verification against a pre-enrolled biometric signature profile database, you can guarantee that a document or transaction can be signed only by the right person. This not only greatly reduces fraud but also dramatically increases the evidentiary weight. Well-known examples here are client authentication for bank transactions and management/staff authentication for high-value purchase orders.

As such, an electronic signature verification system uses all recorded biometric data (speed, acceleration, and pressure), and the false acceptance/rejection rates the system is able to achieve are much better than when simply comparing two or more signature images. Important here is that the pre-enrolled profile stays up to date with natural shifts in signing habits over time. In addition, signature capturing has the advantage of lacking the invasive nature of other biometric authentication methods such as fingerprint, face, or retina scanning. A signature, even if hacked, is not reusable since no one can ever sign the same way twice—signatures are bound to be different from one another. In addition, the signer can always change a signature to create a new personal profile. By contrast, fingerprints etc. do not change (they are static) and may be used again and again.

Additionally, some European countries (for example, Italy) even allow this verification technology based on biometric signatures to be used instead of a numerical PIN to access a qualified personal signing certificate that is stored in a central high security module (HSM). In this case, users can execute a qualified electronic signature (QES) solely with their handwritten signature.

4 Devices Options for Capturing Biometric Signatures

The typical business process as a whole for e-signing in branch offices, retail stores, and customer centers pretty much differs from use cases in which mobility is a central factor. Consequently, devices are often larger so that they can also show documents more conveniently. Additionally, other factors such as running advertisements during idle operation and the possibility of running questionnaires to obtain client feedback is often critical. The most important requirements typically found for e-signing at the POS are listed below.

4.1 Flexibility to use signature pads from the manufacturer(s) of your choice

The type of signature capturing device that fits best is primarily defined by the specific use case and environment condition at hand. The market itself offers an extremely broad range of devices, including very basic signature pads with a b/w display, signature pads with color display, smartphones, pen-enabled screens with a display size of 10” or more, and tablets running iOS, Android, or Windows.

A device-independent solution offers the necessary flexibility. Thus you can integrate the solution using the capturing device that fits the needs for each of his use cases best. This is best addressed with a modular architecture that enables the introduction of new signature capturing hardware through plug and play. Ideally, you can even exchange all the devices you are using today with newer devices released tomorrow without needing to redo your custom integration of the e-signing solution.

This enables companies to avoid being beholden to signature hardware manufactures and lets them make an informed decision each time they need to replace the existing hardware infrastructure. In addition, market experts foresee a great deal of consolidation and new entries in the signature capture hardware business over the next few years. The likelihood that the signature capturing market looks the same as it does now is close to zero.

4.2 Show the whole document

Many use cases, and in some countries even the law, require that the signing device not only display a simple signature box where the client should and does sign, but also the whole document content too. Displaying the document sector that a signature field overlays as background can be also achieved using black and white signature pads, but browsing the entire document and enabling users to read certain paragraphs really requires color devices with good resolution displays.

4.2.1 Signature pads

It is already possible to show the document to be signed on a signature pad with a color LCD of 4–5” given that it provides a high enough resolution. This is basically true for many models, including Wacom STU-530, SIGNificant ColorPad 6, StepOver naturaSign Flawless Pad, and so on. To overcome their limited display size, the devices allow you to scroll the document on the signature pad, either autonomously or through communicating with the e-signature software running on the host PC (desktop). As outlined in chapter 6.2, the response time of the data transmission has to be considered.

4.2.2 Signature screens (pen displays)

Signing on pen displays that typically have a size of 10” or higher and that are used as a second screen absolutely requires e-signature software that manages them appropriately; otherwise, you will not benefit from all their strengths. Windows for example uses the pen display as a desktop extension that uses a stylus as an additional input device. Every time the client touches the screen with the stylus, the focus is shifted by the operating system to the pen display (second screen),

disturbing the operator on the main screen. In addition, it's difficult to train operators to display a document that is ready for signing on the screen area of the second (extended) display.

One clear advantage of signature screens is their instant responsiveness, which is pleasantly different from the rather slow response time of color signature pads. Screens also work great for showing videos and high-resolution images, which is excellent for running commercials when the screens are idle.

However, in a typical setup, you use the signature screen in parallel to the main screen of the operator. Moreover, the operator simply might not see what is shown on the signature screen. Thus, you need to take care of the following:

- When the client reviews and signs a document on the signature screen, the operator must be able to use his or her screen in parallel without being blocked by the client's interaction with the e-signing application. Thus, the e-signature solution must prevent the signature screen from grabbing the mouse focus from the main screen.
- What is shown on the signature screen versus the operator screen needs to be fully automated because having to move application windows around manually on two different screens is simply too big a hassle.
- The operator should see what the clients are doing on their signature screens, allowing the operator to guide and assist the clients by using a monitoring window on the main operator screen.
- Interactive screens are great for collecting customer feedback. Therefore, the e-signature solution should be able to present surveys to the client and collect the answers after customers have completed the transaction.
- When the signature screen is in idle mode, it should show predefined ads such as presentations or. This advertising mode should not interfere with other applications running in parallel on the operator's connected computer.

4.2.3 Multipurpose tablets

Mobile tablets such as the iPad, Galaxy Note 10, or Surface Pro are primarily built for a mobile use case. However, as they can be used for multiple purposes, provide a rather large screen that allows comfortable display of full page documents, are fairly cheap owing to their mass production, and are easily available, mobile tablets are also very interesting for a point-of-sale process. If the sales agent does not work off a fixed desk but has to be somewhat mobile, these tablets are even more useful.

An additional advantage is that these multipurpose devices can be turned into biometric signing devices through a native application that can also be used to cache data, making the devices independent from network connections, bandwidth issues, and/or slow server response times (see chapter 6.2). Additionally, they are ideal if you want to allow users to work with documents like on paper, as described in chapter 2.4.


Furthermore, it is highly beneficial if the signing application on such tablet devices is tightly integrated with the overall e-signing solution, which can also be used with other signature capturing devices such as signature pads and screens. Only then is a mixed infrastructure that includes switching between signature pads and mobile devices, depending on the use

4.3 Using a smartphone to capture biometric signatures

Smartphones, meanwhile, have achieved incredibly impressive market penetration. Nearly everyone has one. As such, why not use them for capturing handwritten signatures and their biometric data—especially in situations where you cannot equip the salesperson with special purpose signature pads, screens, or tablets? You may not want to equip independent sales agencies with such devices, but you can count on every salesperson in that organization having a smartphone that can be used for signature capturing—so let's use them.

All you need to do is providing a small biometric signature capturing app on the smartphone that is compatible with the back-end component of your e-signature software. Simply sign with a capacitive stylus, a finger, or with the native pen should the smartphone come with one.

The typical process includes the following:

- Review documents or complete form fields and then add attachments on any computer in the browser—perhaps together with a customer, employee, or business partner—and use a smartphone as a signature capturing device
- A native app turns a smartphone into a signature capturing device. This app should be available for most iOS, Android, and Windows Phones.
- When the signer is ready to sign a document, a secure communication between the smartphone and the host computer is established.
- The secure communication is done through a server application using a token. Therefore neither the host computer nor the smartphone need to be reconfigured—both can simply use their existing network connections. The token can be read for example with the smartphone's built in camera using a QR code reader integrated into the native signing app.

- The signature app is showing a signature capture dialogue with the document background, providing a visual document mapping.

- The signature is captured on the smartphone. It’s highly recommended to use smartphones with native pens or a stylus for signing; otherwise, you may lose the potential for forensic identification.
- After the signature is captured, it is transferred to the host computer via the secured channel and then embedded into the document.

5 Providing Evidence

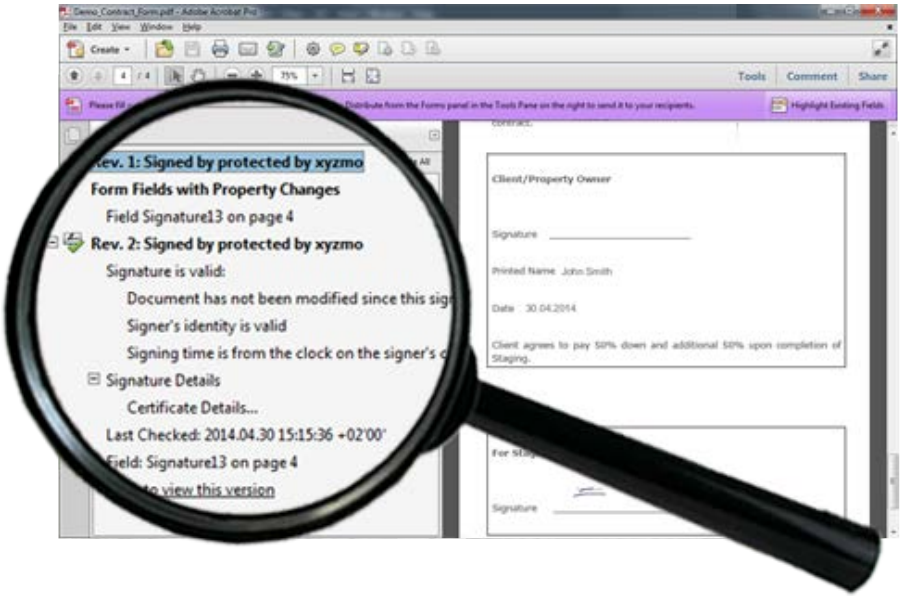
Proving the authenticity of an e-signed document depends on the audit trail that the e-signature solution, which has been used to sign the document, provides. This audit trail can either be stored in the document itself, enabling the document to be self-contained - see chapter 1.2 - or separated, or a combination of both.

Audit trails can also do much more. A proper audit trail that includes authentication results for the signers shifts the burden of proof toward the signer in a court proceeding, especially if the solution has processed a lot of documents already without problems. The judge will automatically have a legitimate expectation that the solution also worked for the document in question. It is important that the audit trail is understandable by the involved judge and lawyers without the need to consult a technical expert for interpretation. If the user actions have also been logged, then this can be used as process evidence and further increase the evidential weight.

Note: Particularly if you use cloud-based solutions, you have to be sure that you have everything you need to proof the authenticity of documents many years later, even if by then you are no longer a client of the vendor or the vendor simply does not exist anymore.

5.1 Evidence provided by a digital signature/certificate

By reviewing the digital signatures in a PDF document, you can look at the embedded signature history within standard-compliant PDF viewers, even if you are not connected to the Internet. This way, you can see exactly what the document looked like when each digital signature was applied.



Additionally, the digital signature also provides evidence about the following important aspects:

- The document's integrity (see chapter 3.2),
- The date and time the document was signed—optionally through a trusted time stamp service,

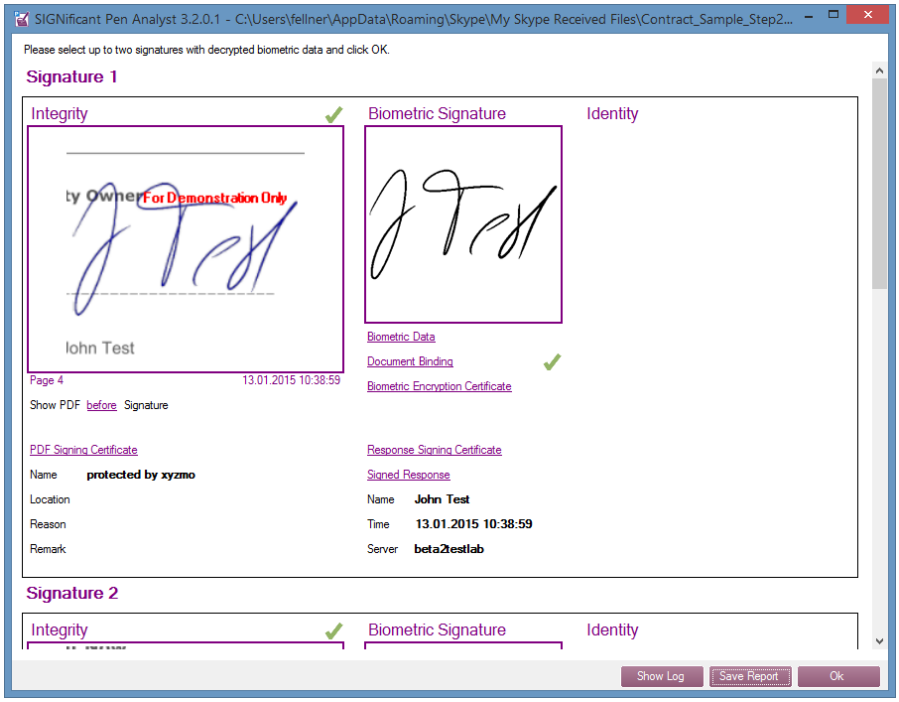
- The geolocation where the document was signed (GPS data if provided),
- The issuer of the signed document (through the used digital certificate).

5.2 Evidence provided by biometric signature data

The biometric signature allows you to identify who has signed the document without any additional server-based audit-trails (see chapter 5.3). However, this kind of evidence requires:

1. That it is possible to decrypt the signature data from the document, which can be done using its securely stored private decryption key,
2. That, as in the paper world, a signature expert (graphologist) is able to do a manual signature verification.

The second bullet point may not be necessary in case you can provide evidence that the recorded biometric signature data has been reliably verified in real-time against known sample signatures of the signer before being embedded in the document. To provide trustworthy evidence, the biometric signature must be linked with a signed response to an identified signature verification server (see chapter 3.4). This signed response must be digitally signed to make sure that the system is not vulnerable to bypassing it (e.g. through a hijacked verification service). This way, you can easily prove that the recorded verification results have been provided by a successfully authenticated and certified verification system and thus provide evidential weight.



Signature audit trail incl. a signed biometric verification response of a self-contained PDF document

5.3 Typical server-based audit trails

Server-based audit trails can be stored independently from the signed document (e.g. in a central location), which may simplify document archiving and distribution.

5.3.1 Action log

Server audit trails develop their full potential when they provide process evidence such as logging document distribution to a specific point-of-sale or field agent and the executed actions within a document. Audit trails should document what happened to a specific document, in what order, at what time and where. This can include confirmation of pages where special tags have been set which forced the signer to confirm that he or she read the marked sentences. The log should at least keep track of the accomplished tasks that have been performed by the signer based on the predefined guiding through the document (see chapter 2.1). Certainly, detailed information about executed authentication steps such as typically required with HTML5 signing (e.g. SMS-TANs sent to registered phone numbers, or scanned IDs) are most critical.

5.3.2 Biometric real-time signature verification audit trail

In case you want to store the captured biometrical data only in a central location - as opposed to storing them in the PDF documents as well - you may simply reference it from the action log (see chapter 5.3.1), using the identifier (RequestID) of the performed verification request, to the audit log of the signature verification server (see image below).

Additionally, as this veification audit trail does not include the biometrical data itself but only references them, the signature authentication proof is much more accessible because access to it (e.g. the ability to show it to a judge) does not require its decryption using the private key that you need to extract the biometrical signature data from a PDF.

Audit Trail für Profil:default

RequestId	Datum/Zeit	Tätigkeit	Ergebnis	Profil	Geräteprofil	Unterschrift	Beschreibung	Authentifizierter Benutzer (IIS)	Angaben zum Host
40d386bc-b4d7-475f-bca1-fd02f0094fd2	Friday, July 04, 2014 9:58:40 AM	DeviceProfileAdd	Ok	default	ColorPad/SIGNificant ColorPad		Geräteprofil hinzugefügt	sign043\usermanager	SIGN043 (4.3.0.8)
40d386bc-b4d7-475f-bca1-fd02f0094fd2	Friday, July 04, 2014 9:58:39 AM	VerifyUserProfileDynamicToDynamic	VerifyMatch (96%)	default	Samsung GalaxyNote 8	NaturaSignPadColor		sign043\usermanager	SIGN043 (4.3.0.8)
8a0a8819-13b5-416b-8bc7-5a2ed9baaef4	Friday, July 04, 2014 9:58:31 AM	VerifyUserProfileDynamicToDynamic	VerifyNoMatch (79%)	default	Samsung GalaxyNote 8	NaturaSignPadColor		sign043\usermanager	SIGN043 (4.3.0.8)
82cdfd1b-dc82-450f-b777-2086b04265be	Friday, July 04, 2014 9:58:22 AM	VerifyUserProfileDynamicToDynamic	VerifyNoMatch (67%)	default	Samsung GalaxyNote 8	NaturaSignPadColor		sign043\usermanager	SIGN043 (4.3.0.8)
3daaaae5c-d3a9-4c94-9f03-f4fbe8770f59	Friday, July 04, 2014 7:36:29 AM	VerifyUserProfileDynamicToDynamic	VerifyMatch (86%)	default	Samsung GalaxyNote 8	samsung GT-N5110 JZO54K		sign043\usermanager	SIGN043 (4.3.0.8)
ed67234a-e189-4cc4-a846-5616b6085b3b	Friday, July 04, 2014 7:36:13 AM	DeviceProfileAdd	Ok	default	Samsung GalaxyNote 8		Geräteprofil hinzugefügt	sign043\usermanager	SIGN043 (4.3.0.8)
ed67234a-e189-4cc4-a846-5616b6085b3b	Friday, July 04, 2014 7:36:12 AM	EnrollDynamicContinuous	EnrollContinued	default	Samsung GalaxyNote 8			sign043\usermanager	SIGN043 (4.3.0.8)
ab30ccd2-b1c2-4763-9e37-4510cff8adf9	Friday, July 04, 2014 7:36:12 AM	ProfileAdd	Ok	default			Profil hinzugefügt	sign043\usermanager	SIGN043 (4.3.0.8)

Using such an audit trail of executed real-time signature verifications that is easily readable by non-technical person and by a non-product expert (such as a judge or counselor), as shown in the image above, and the signed response data of an executed verification stored in the signature field of a signed document (see chapter 0), you can greatly increase a signature's evidential weight and reliably prove that only an authenticated and documented person was able to sign a specific document. Thus, the

burden of proof that the document was not signed by this person is more or less now put on the signer himself or herself (= reversal of burden of proof).

6 Integration and Deployment Requirements

In addition to the functional requirements to the e-signature solution itself that have been introduced in the previous chapters, you will face requirements that deal with the integration into your existing IT and application environment. The most important ones are discussed in this chapter.

6.1 Standalone GUI App or SDK

If you require a fast and cost-efficient deployment, a ready-to-go graphical user interface is typically the best choice. This option usually still allows easy customization of color schemes, logos, etc. to your requirements.

If you do require a seamless integration into an existing application (without a UI context switch) then the SDK approach will be the right one. Here you can manage the detailed user experience and all GUI elements through advanced coding yourself. Powerful SDKs allow much more than simple integration of core functionality – such as providing a complete adaptable user interface with a framework to seamlessly integrate it.

Thus, the more functionality the SDK includes from a full application, the better it is. Powerful SDKs even include an application's full user interface and simply offer its parameterization, which can be typically done in a few days versus the weeks and months that are needed when you start from a low-level SDK that only offers some core functionality such as signature capturing.

An important disadvantage when using a **low-level SDK** that just exposes the necessary signature capturing and document manipulation functions is that the company's IT department will need to manage many of the security questions itself (e.g., protecting customers' handwritten signatures from unauthorized access). This is an enormous burden on the IT and compliance departments, as it is not their everyday workload, resulting in increasing sources of errors. This includes efforts to avoid misuse and careless coding, making the task of providing a sufficient level of security even more difficult. The fact is that even if all security precautions are implemented, this issue is usually not easily explained to end customers and third parties, as theoretically, the company/the employees can possibly misuse this sensitive data. Thus, a powerful SDK that wraps an entire standard application is a much better choice because this way the company can easily prove that it could not possibly manipulate signatures.

6.2 Fast operation in low-bandwidth environments

Questions about response times and bandwidth requirements between client and server become important particularly when a server-based architecture is being deployed. Here, server-based solutions can minimize their bandwidth requirements through local caching and background synchronization.

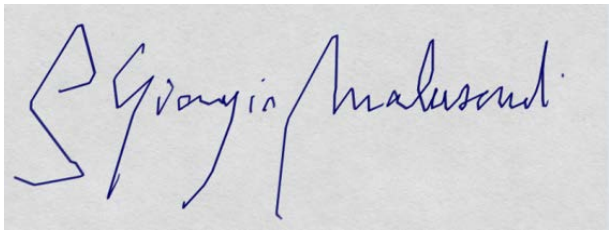
Response times are dependent not only on server performance and on scalability but also on the response time of the signature capturing device. Although tablets with native apps

and signature screens by design work with nearly no delay, this is not the case with USB-based signature pads. This is because signature pads are peripheral devices that only display the content they receive through their USB connection—typically as images. The typical response time of signature pads with color display is about 2–3 seconds for transmitting the data from the host PC (desktop) to the signature pad.

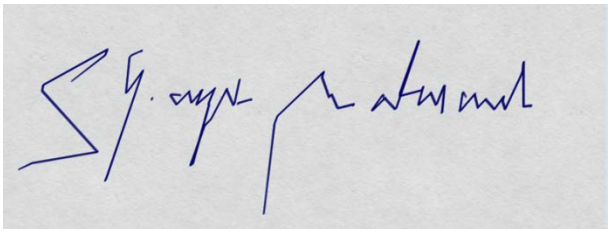
6.3 Enable thin clients to use USB signature capturing devices

In case your POS client PCs are virtualized with Citrix, VMWare, or Windows RDP/Terminal Services, your e-signature software needs to locally buffer the data recorded by the USB signature device on the thin client; otherwise, some of the captured biometric data packets will be lost due to network latency. This is because signature pads send the data they record with fire-and-forget to a local buffer, which is not an issue as long as the software, which reads and edit the data, runs locally. However, in a thin client environment, the buffer that stores the received biometric data packets might not be read in time, because access over the network is delayed by the its latency. Thus, a simple pass-through does not work.

The illustrations below show how network latency influences the quality of signature capturing on a thin client terminal without a local software component to take care of correctly receiving the data packets from the local USB signature capturing device:



60 ms latency



100 ms latency

6.4 On-Premise vs. Cloud

When you decided on a client/server architecture (see Chapter 1.3 for the pros and cons) you have to choose a deployment model for your back-end infrastructure. It is possible either to get the solution and run it in a private cloud, consume it through an SaaS model or to deploy and run it on your own premises.

Whereas the cloud model is faster and easier to set up, and also typically provides a limited option to define where your servers and data should be located, the on-premises option is still preferred by many organizations. This is because all applications and files are located within your private data center, which means that you are consequently not dependent on external systems or on Internet issues. Additionally, only the on-premises model offers you full control over data privacy, something that cloud services simply cannot guarantee.[1]

[1] http://www.zdnet.com/how-one-judge-single-handedly-killed-trust-in-the-us-technology-industry-7000032257/

With an on-premises model, you simply need to choose between a native installation, in which the software runs natively on the computer, and a virtualized approach, e.g., using VMWare, Citrix, or Microsoft virtualization technologies.

7 SIGNificant-References

SIGNificant is an enterprise e-signature platform that allows you to go completely paperless at the point of sale (POS), regardless of whether it is in a direct branch office or shop, or through an indirect sales channel that you cannot equip yourself. SIGNificant provides you with the user interface and tools needed to define an optimal e-signature process and user experience. Whether for signature pads, interactive pen displays, mobile devices, or Web-based signing, the platform's building blocks make it easy to select the best combination of e-sign solutions and signature capturing devices for each use case.

To better illustrate how SIGNificant can be applied in selected industries for their specific use cases in POS environments, the following section outlines real case studies including their end-to-end business process that has been implemented.

7.1 Retail banking: GE Money Bank (Czech Republic)

Use case:

- client bank transactions (deposits, withdrawals, transfers)
- standard contracts (account opening, credit card, etc.)
- loan contracts and agreements
- financial investment contracts

Deployed products:

- Signing application: SIGNificant Server with Web Signing Interface and Linux-based Citrix Components on Dell Thin-Client Terminals.
- Authentication application: SIGNificant Biometric Server—Enterprise Edition with Oracle signature database.
- Signature capturing hardware: SIGNificant ColorPad 6.

End-to-end business process:

1. The client goes to the branch and is welcomed by an employee.
2. If the client is not yet enrolled in the signature database, the client authenticates to the operator using an identity card (e.g., national ID) and enrolls in the SIGNificant signature database.
3. The operator processes the client's request (e.g., cash withdrawal).
4. The client reviews the document to be signed directly on the SIGNificant signature pad and signs it directly with his or her handwritten signature on the pad.

5. The SIGNificant Biometric Server verifies the handwritten signature in real time against the bank client's signature profile stored in the database to execute an authentication check on the transaction.
6. If the result of the authentication is positive, the request is processed, and the SIGNificant Biometric Server signs the transaction document with the captured biometric signature data and then digitally seals it with a trusted time stamp and a certificate managed securely inside the HSM of the bank.
7. The system puts the signed PDF/A document into a legal archive.
8. Nothing is printed unless the client strongly wants a paper copy.
9. The client can access the signed doc on the Web application.

7.2 Retail market: REWE Stores (Germany)

Use case:

- Digitally sign electronic debit process receipts and credit card receipts on self-checkout points with a handwritten signature.

Deployed products:

- Signing application: SIGNificant Server with Cash Register Plugin
- Signature capturing hardware in shops: Wacom STU-500

End-to-end business process:

1. The client goes to the checkout point in the store and registers his or her goods for checkout.
2. The client selects to check out with either electronic debit process or credit card.
3. The client reviews the final bill, time and date, and payment method on the screen of the Wacom STU-500 signature pad and directly signs on it with a handwritten signature.
4. The SIGNificant server signs the document with a handwritten signature and then digitally seals it with the REWE signing certificate.
5. Nothing is printed unless the client wants a paper copy.