muhimbi

Making InfoPath Documents Available for Long Term Archiving

The Need for Conversion

This document describes the challenges that organizations who utilize Microsoft InfoPath will encounter if they attempt to open the same InfoPath documents in the future, be it 1 year or 20 from the date the form was created.

Some background is provided about why this may be problematic, as well as a number of potential solutions.

Intended audience of document

The intended audience is anyone involved in the development and maintenance of InfoPath forms. This document should be of particular interest to IT Management, especially in organizations subject to strict regulatory requirements.

DISCLAIMER

muhimbi

TABLE OF CONTENTS

Chapter #1
About this white paper

This free white paper is provided by Muhimbi Ltd (muhimbi.com), home of the PDF Converter for SharePoint. Feel free to contact us at support@muhimbi.com, we love talking to our customers.

EXECUTIVE SUMMARY

Microsoft InfoPath is a simple yet powerful tool to create forms. As a result almost anyone can create a centralized form for entering data. However, without spending a significant amount of time on planning and architecture, it will most likely not be possible to unlock the information stored on these forms in the future. For practical, legal, fiscal, or regulatory[1] reasons this may be problematic.

Even with careful planning, it is best to accept the inevitable and choose a different file format to archive the contents of the InfoPath forms once they have been through their life-cycle. Once archived in a different file format such as *PDF* or *MS-Word*, the ability to open documents years down the line is guaranteed, with the added advantage that the content of forms can be easily shared.

If your InfoPath data is non-trivial and it is important to maintain access to the data stored in your forms, then act before it is too late.

The Muhimbi PDF Converter for SharePoint makes it easy to automate the conversion of various file formats such as InfoPath, including all attachments, to file formats more suitable to archiving such as *PDF, PDF/A1b, PDF/A2b, MS-Word, Excel* and *HTML.*

ABOUT THE AUTHOR

Jeroen Ritmeijer

Jeroen is CTO and one of the founders of Muhimbi Ltd, a provider of document conversion products for the SharePoint market. Before Muhimbi, Jeroen held several CTO and Technical Architect positions in the finance and retail industry. As an architect of third-party products he has a technical knowledge of SharePoint and InfoPath that extends well past most developers.

muhimbi

---

[1] Many regulatory bodies issue document retention regulations: *SEC, FTC, FCC, EPA, NLRB, IRS, EEOC, OSHA,* and *OFCOM* are some examples. The *Sarbanes-Oxley Act (SOX)* requires a 7 year retention level, *HIPAA* requires a minimum of 6 years while *CMS* requires a minimum of 10 years. *The Freedom of Information Act* requires data to be easily accessible for periods between 2 and 10 years

Chapter #2
What / Why InfoPath

InfoPath is a simple yet powerful tool to create User Fillable Forms. Forms are stored in a central location after which they can be filled out by users throughout an organization. Read on for details about how InfoPath came to be, a description of how it is generally used in combination with SharePoint, as well as some examples of how people and organizations use InfoPath.

HISTORY OF INFOPATH

The first version of InfoPath was released as part of Microsoft Office 2003. This lead to some confusion as its purpose was not immediately clear and few people knew what to do with it. It was described as an XML Editor by the organizations that were trying to sell it.

To put this into perspective, *XML* was a common buzzword for the first few years of the new millennium. Organizations were scrambling to leverage XML by creating XML editors. As a result, many low level XML Editors were created by various organizations. Similarly, Microsoft released InfoPath as a tool to hide the complexity of XML with a specific focus on entering data into forms. People who design InfoPath forms are probably aware that the underlying file format is XML, but those filling out the forms are probably not aware—which is a good thing.

InfoPath matured with the release of Office 2007 and Office 2010. Obvious shortcomings were addressed and Microsoft's famous *Ribbon Bar* was included in the 2010 version.

With the release of InfoPath 2007 the product started to take off in conjunction with the release of SharePoint 2007, an excellent platform for storing and filling out forms. Microsoft Forms Server, an add-on included in the SharePoint Enterprise Edition, allows InfoPath forms to be filled out using a standard web browser.

InfoPath only received minor updates as part of the Office 2013 release, a sign of maturity for the platform.

WHERE DOES SHAREPOINT COME INTO THE EQUATION?

If it hadn't been for the release of SharePoint 2007 and its integrated Forms Server InfoPath would probably have died a silent death. SharePoint is the ideal companion product for InfoPath and leverages it in a number of ways:

1. SharePoint is the ideal location to store the Shared Form (aka XSN file).

2. InfoPath can link to SharePoint lists and other SharePoint data sources.

3. InfoPath is used to design forms for Microsoft Forms Server, the browser-based form editing service that ships with SharePoint 2007, 2010 and 2013.

4. InfoPath forms can participate in SharePoint Workflows and become part of a larger business process.

5. InfoPath data can be exported to a SharePoint column so its contents can participate in views, workflows and filters.

HOW DO ORGANIZATIONS USE INFOPATH?

Organizations use InfoPath because it makes it easy to capture structured data. Capturing structured data is not exclusive to InfoPath, as it is a typical use case for a Relational Database Management System (SQL Server, Oracle). However, InfoPath also controls how that data is presented to the user.

Common uses for InfoPath are for the capture and maintenance of the following:

- Customer / Patient data

- Employee performance reviews

- Expense forms (including scans of expenses)

- Asset tracking

- Change requests

- Invoices

- Bug / Issue tracking

- Meeting agendas

- Purchase requests

This is merely a short list of examples. If there is a form for a task (ask your local government), then it can likely be modelled in InfoPath.
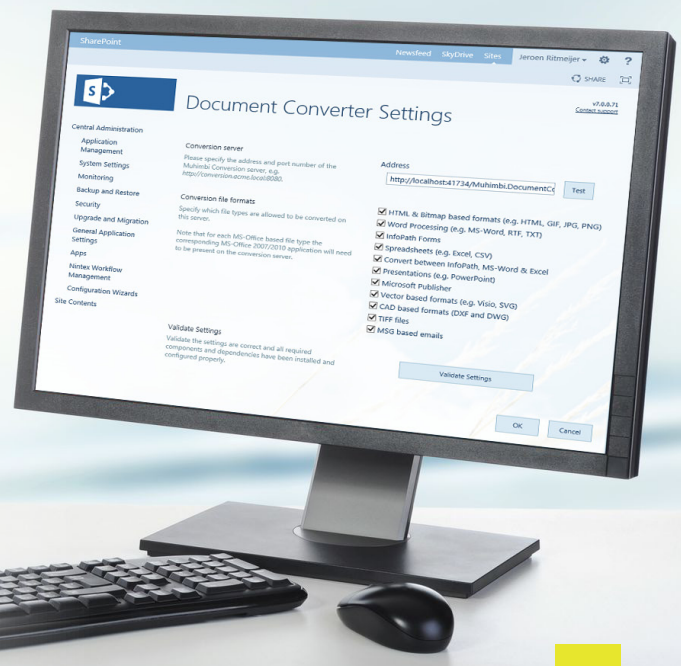
Chapter #3
InfoPath 'Under the hood'

Although it is relatively easy to get started with InfoPath, it is important to know what is going on *under the hood* as InfoPath is both a *Document Type* and a *Development Platform.*

IT IS A DOCUMENT TYPE

InfoPath is a **document type** in the same way that MS-Word and Excel are document types– you need InfoPath to open these documents. It provides a relatively rich environment for designing forms, even making it possible to create different views for presenting different data and to embed files.

The fact that InfoPath is a document type frequently leads to confusion as many users think that the XML file saved by InfoPath is similar in function to the DOCX file saved by MS-Word. It is not unreasonable for users to assume that they can simply take this XML file and email it to someone else. Unfortunately this will rarely have the desired effect. An InfoPath XML file merely contains the data associated with the form, not the *look-and-feel* of the form which is stored in a separate XSN file. This leads to the next topic.

IT IS A DEVELOPMENT PLATFORM

One of the key advantages of InfoPath over other file formats is that InfoPath is also a development environment, similar to Visual Studio being a development environment.

1. Rich User Controls such as date-pickers, drop-down menus and file attachment controls can be added.

2. InfoPath comes with an integrated Rules / Macro language for simple automations.

3. More complex "code-behind" logic can be added using C# or VB.net.

4. InfoPath integrates with SharePoint List Data.

5. InfoPath integrates with third-party databases such as SQL server.

6. InfoPath integrates with third-party web services.

7. InfoPath uses a strong security model to prevent unauthorised code from running.

InfoPath is used as a development platform by a power user or developer to develop a form, aka an XSN file. It is this centrally stored XSN file that contains the look-and-feel, together with the XML file mentioned in the previous section it makes up a single form. Let's illustrate this in an example:

BUILDING THE FORM

Throughout the next few sections we will use a simple Holiday Request Form as an example. The point of this exercise is not to provide a detailed step-by-step guide about how to create InfoPath forms, but rather build a foundation for the rest of this white paper.

In this example we use InfoPath Designer to build a Holiday Request Form for a fictitious, and rather unusual, organization. This is a light-hearted and perhaps somewhat convoluted example, but the importance of the various elements will become clear later.

Once designed, the form is published to a central location in SharePoint as an XSN file. **The XSN file contains the various elements that make up the form created in the example on the right.**

- Look-and-feel defined as XSL-based views.

- References to external data sources.

- An XSD Schema describing the various fields, data types etc.

- A manifest that ties it all together.

Example – Part 1

The form consists of the following parts:

1. Requester: A drop-down box that pulls in a list of all employees from a SharePoint list. The person associated with the holiday request selects an option from this field.

2. Start and **End dates:** The period covered by this holiday request.

3. Essay: No one said that this is a particularly sane organization. The person requesting the holiday needs to upload an essay explaining why they deserve the holiday.

4. Proof of previous holiday: To make things worse, a photo from a previous holiday needs to be uploaded, proving that the requester knows "how to have fun."

5. Status: The *Approval Status* field can be exported to a SharePoint column so it can be queried as part of a workflow.

If custom code was added to the form then this code is compiled and DLLs are embedded in the XSN file.

The important takeaway is that a single XSN file is published to a shared location, and all filled-out forms refer to the location of this XSN file.

Example – Part 2

FILLING OUT THE FORM

To fill out a form, the user navigates to the Forms Library in SharePoint and selects the option to create a new form. If Forms Server is installed, then this will open a browser window. However, in our example Forms Server is not installed so a desktop application named *InfoPath Filler* is opened on the user's system.

InfoPath Filler loads the centrally-stored XSN file to determine how to display the look-and-feel of the form. It also sees that it needs to retrieve the list of employees from a SharePoint list.

With the form fully displayed in *InfoPath Filler*, the user enters the data and attaches associated files. When the form is submitted, it is important to remember that only the user-entered data and a reference to the XSN file are saved. The XSN file itself is not modified and no data related to the look-and-feel, or forms-related logic, is saved into the individual XML file.
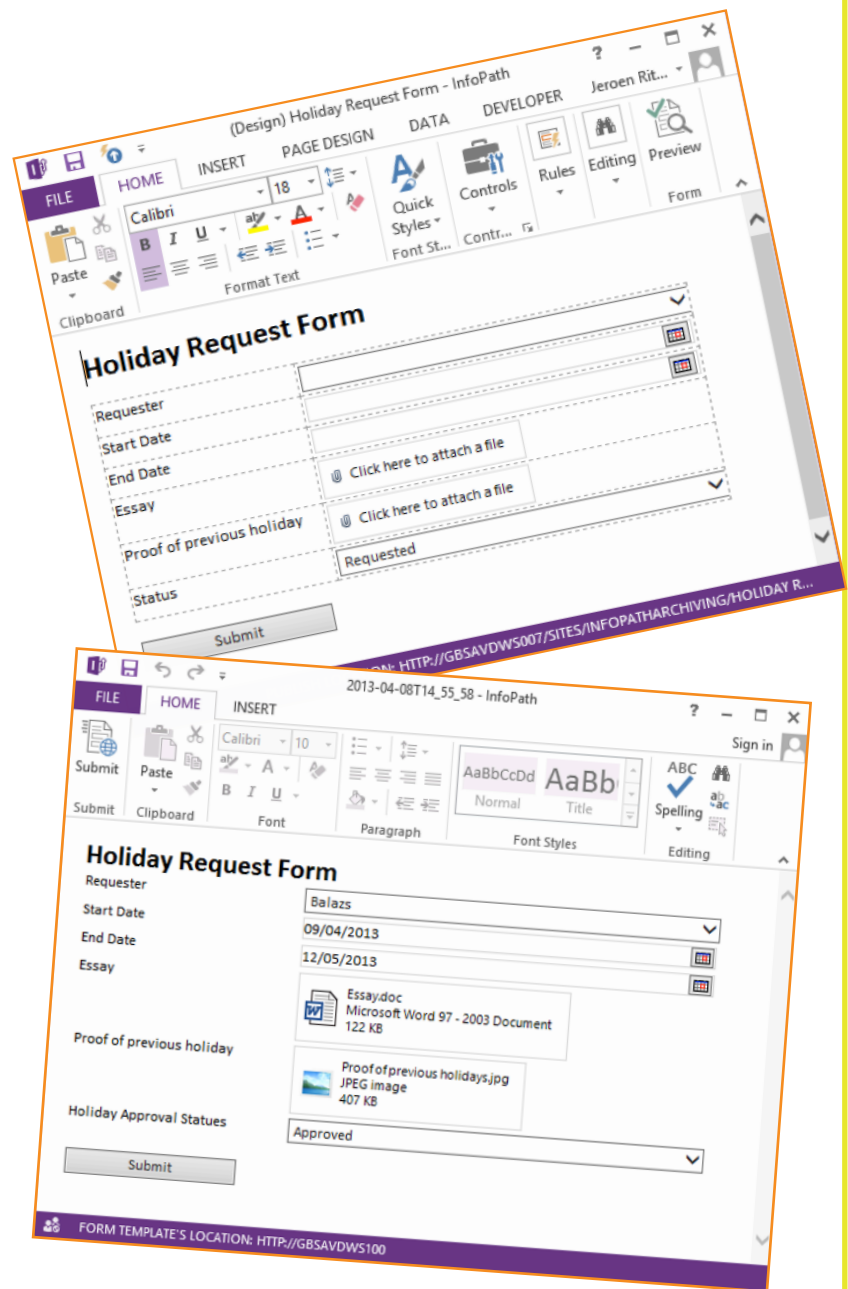
Chapter #4
The Problem

In the previous section we showed how to create a simple InfoPath form and use *InfoPath Filler* to fill out the form and save the data. Where's the problem? For practical, legal, fiscal, or regulatory reasons (HIPAA, SOX, CMS, etc.), most organizations need to make sure that their data is still accessible years in the future. When it comes to MS-Word or PDF files, this is pretty much guaranteed. However, due to the complex nature of InfoPath documents it is unlikely that you will be able to open your documents even a year from now.

You may also want to share your forms with people or parties outside the organization, or people without access to SharePoint and all dependencies needed to display the form. This is not possible without converting the form to a more commonly used file format.

USE OF 'VOLATILE' DEPENDENCIES SUBJECT TO ENTROPY

Almost all processes in nature become the victim of entropy[2], the inevitable and steady deterioration of a system declining into disorder. The same can be said about IT environments– even those that are highly structured and maintained by disciplined staff.

InfoPath forms are not self-contained like MS-Word or PDF files. **They depend on a number of external factors, such as:**

1. Location of the Shared XSN file. First and foremost, all InfoPath forms refer to the Shared XSN file. When your organization migrates from SharePoint 2007 to 2010 to 2013 or some cloud based solution, the exact URL to this shared XSN file will almost certainly change.

2. SharePoint Data Sources. External data sources such as the *Employee List* in our example will need to remain unchanged and in the same location. Fields can't go missing or be renamed. InfoPath allows the location of these data sources to be abstracted, which is yet another thing that will need to be maintained over the years.

3. Database Connections. Similarly to SharePoint Data Sources, any data sources that refer to databases will need to remain unchanged. The same goes for the underlying database tables, field names and data types.

4. Web Service Connections. InfoPath forms communicating with external Web Services will need to remain unchanged over the years. Let's hope the vendor does not go out of business.

5. User Rights and Privileges. The person opening the form must have network access to—and be authorized to access–all of the above-mentioned dependencies.

---

[2] "Just as the constant increase of entropy is the basic law of the universe, so it is the basic law of life to be ever more highly structured and to struggle against entropy." - Václav Havel

Unless the planets align and the *dependencies*, *systems*, *locations*, *network access*, *user accounts* and *privileges* remain unchanged over the years, there is little chance you will be able to open your InfoPath forms far in the future.

Accept the inevitable entropy. Archive your InfoPath forms to a more durable format such as PDF or MS-Word.

RESTRICTIVE SHARING WITH 3RD PARTIES

While the availability of your InfoPath forms in the future is an important topic, it can be equally important to effectively share forms with people external to the organization or those without access to a specific part of the network. This can be difficult since forms can only be viewed by those with access to all dependencies listed in the previous section.

Share InfoPath forms with parties internal or external to the organization by converting the form to a self-contained format such as PDF or MS-Word.
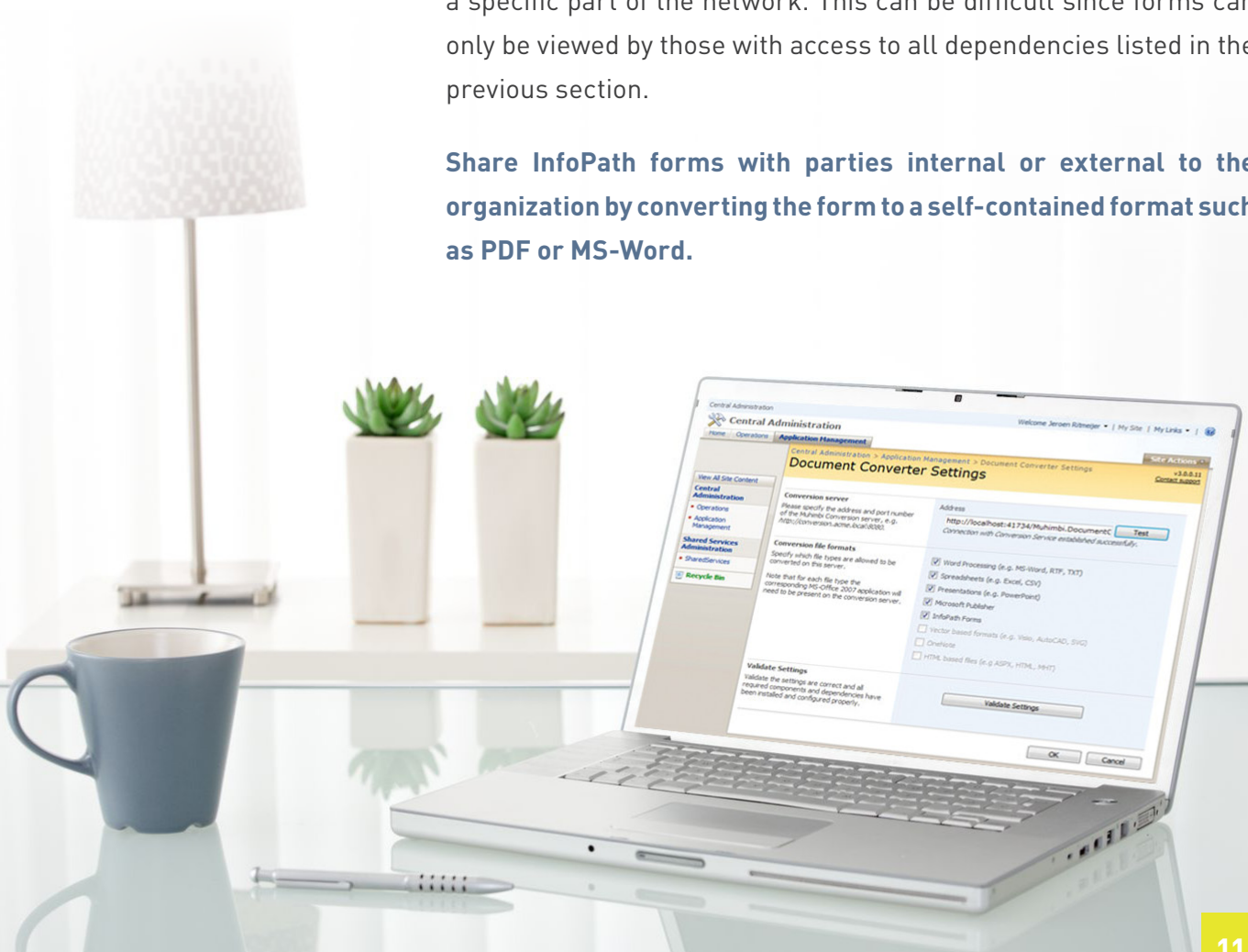
Chapter #5
The Solution

Whether fighting IT Entropy, or sharing InfoPath forms with other parties, a number of solutions are available to archive InfoPath forms in self-contained formats such as MS-Word or PDF.

Solutions range from manual PDF conversion for small teams with simple requirements to fully automated conversion systems that participate in an overall business process/workflow.

MANUAL PDF CONVERSION

Other than, possibly, printing the forms and archiving them on paper, manual conversion to PDF is perhaps the simplest option to implement. Although simple does not automatically equate to good, small teams with very simple requirements, small workloads, and high level discipline may find this to be a suitable solution.

The team working with the InfoPath forms will need to be trained. Every time a user creates or modifies a form, the *Create PDF/XPS* option needs to be selected in the User Interface. The user will then need to browse to the relevant location and save the PDF file.

This only converts the main form - attachments are ignored - but it may be a reasonable solution for some environments.

Advantages of manually converting InfoPath forms to PDF

- It is cheap, provided that all users have InfoPath Filler. This trick does not work in Browser-based environments that rely on Forms Server.

- It is simple to implement.

Disadvantages

- It is not automated. The user needs to remember where to save the PDF and (re) save the PDF whenever a file is modified, requiring a high level of discipline.

- It does not work with Browser-based Forms Server environments.

- User training is required.

- Attachments are not converted.

- The correct InfoPath view is not automatically selected.

- It only converts to PDF. Other file formats are not supported.
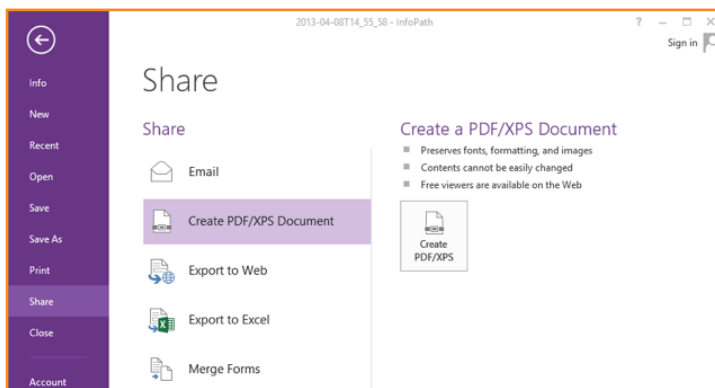
PROGRAMMATIC PDF CONVERSION

The PDF Archiving solution based on manual conversion is simply too restrictive for most organizations. Those organizations may choose to fully automate the process by asking their in-house developers for assistance.

If your organization has developers with sufficient time and knowledge available and you are dealing with a small number of different forms, then there is some merit in going down this road.

Developers can use freely available libraries to programmatically generate files from scratch (for PDF use *iTextSharp,* for DOCX use *OpenXML*). As everything is developed from scratch and by hand, developers have full control over the look–and-feel of the generated file.

If attachments are involved, then it is likely important to convert and merge them, which is difficult to achieve programmatically. Most likely you will need a third-party product such as the **PDF Converter Services*** to programmatically convert attachments into the desired format.
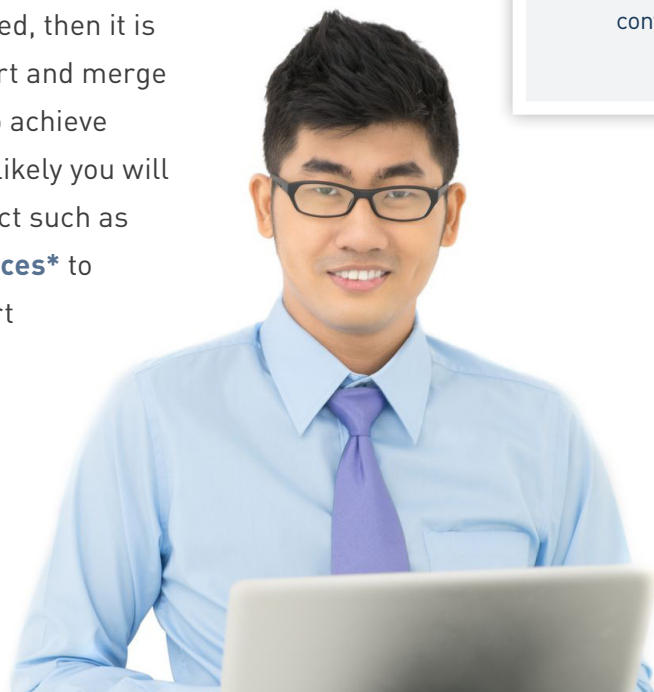
Advantages of converting files using in-house developed software

- It is very flexible.

Disadvantages

- It can be expensive. Developers are not cheap and InfoPath conversion is not trivial to implement.

- The process can be slow and drawn out.

- New code is required for each new InfoPath Form.

- Code changes are required for every change to the Form.

- Additional software is required for conversion of attachments.

* Learn more about PDF Converter Services at: http://tiny.cc/PDFConverterServices

FULLY AUTOMATED THIRD-PARTY ADD-INS

A fully automated solution is the recommended way forward. Custom in-house development is an option, but that can be a slow, lengthy, and expensive process. Fortunately third party software is available to alleviate the pain.*

When using a fully automated process, people who fill out forms do not need to be trained to carry out manual conversion tasks. The right view is selected and all attachments are converted as well. *Total IT bliss.*

In the high-level example below, we'll show how to create a basic SharePoint Designer workflow to convert the InfoPath Form created earlier to PDF. We can achieve the same using Nintex workflow, and the steps are identical in SharePoint 2007, 2010, and 2013. In order to carry out these steps, the Muhimbi PDF Converter for SharePoint will need to be installed on the SharePoint farm.
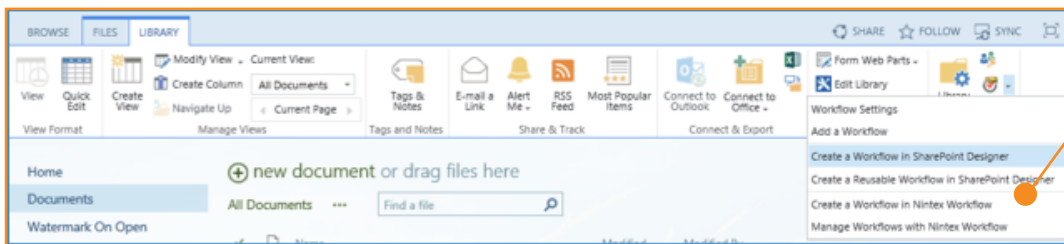

To create a new workflow, navigate to the **Library** used to store the InfoPath forms and select the option to **Create a Workflow in SharePoint Designer** from the Library ribbon.

Provided it is installed, SharePoint Designer is automatically launched and a new workflow is created. Using the *Workflow Settings* button, we instruct it to start automatically when a Form is created or updated.

Using a *Conditional If* statement, we check that the file triggering the workflow is not already in PDF format and that the *Status* field of the Holiday Request Form has been set to *Approved.*

Using the *Actions* ribbon icon we select the *Convert Document Action* and specify which item to convert, the location to write the file to, and which file type to convert to (*PDF, MS-Word, Excel, HTML*, etc.)

* http://tiny.cc/PDFConverterForSharePoint
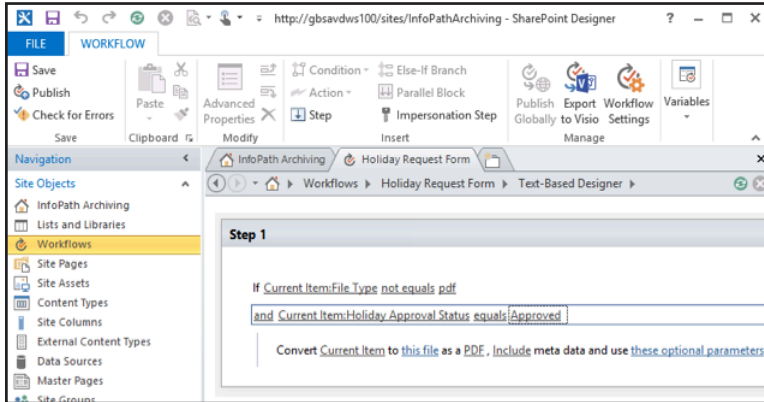
Once the workflow is published to SharePoint, it will automatically execute whenever a Form is created or updated. During a conversion the PDF Converter will automatically convert the correct InfoPath view, convert all attachments, and then merge it all together into a single self-contained file ready for archiving or sharing.

A PDF Reader's scrollbar can be used to scroll through the Form and all attachments, but it is also possible to navigate between the main form and all attachments using automatically generated PDF bookmarks.
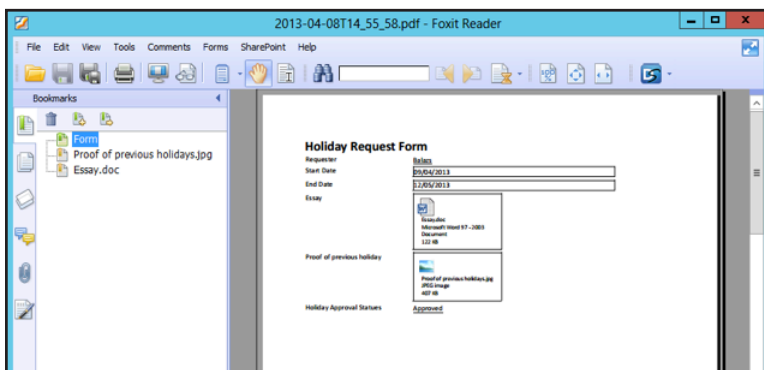
Advantages of the fully automated option using third-party software

- No user training is required.

- Document conversion will never be "forgotten."

- Attachments are converted as well.

- Correct View(s) are converted.

- It works with *InfoPath Filler* and Browser Based forms.

- Third-party software is maintained at no extra cost. New versions of InfoPath and SharePoint are automatically supported.

- It works with a large number of different InfoPath forms

- Changes to InfoPath forms, and new InfoPath forms, are automatically supported without any extra work.

- It does not require expensive, or overworked developers.

Disadvantages

- Like all software, some thought needs to be put into deployment.

Case Study #1

The possibilities are endless. One of our customers uses InfoPath to record incoming orders in combination with a SharePoint Designer workflow to control the life cycle of the form and route it past a number of different departments. As the last step in the process, it uses the Muhimbi PDF Converter for SharePoint to convert the InfoPath form and all attachments into a single PDF.

This PDF is then sent to an outsourcing center in India for further processing. As this is an external company without access to the network resources the InfoPath form relies on, PDF is the perfect file format as it is completely self-contained. Before the file is sent via email, the PDF Converter for SharePoint is used to encrypt the PDF and apply an "Open" password to prevent the PDFs from being accessed by unprivileged users.

> " PDF is the perfect file format as it is completely self-contained."

Case Study #2

A large insurance company in the United Kingdom uses InfoPath for capturing new customer data. At the end of the form's life cycle, the Muhimbi PDF Converter is used to automatically convert the file to PDF/A1b format, a PDF substandard created specifically for the purpose of long term archiving. The PDF/A version of the form is then stored in a records center.

The PDF Converter is also used to work through the initial backlog of 300,000 forms without a single failure.

> " The PDF Converter is also used to work through the initial backlog of 300,000 forms without a single failure."

Muhimbi PDF Converter for Share Point

The Muhimbi PDF Converter for SharePoint (www.muhimbi.com) is the ideal product to automate the conversion of InfoPath as well as other file formats.

Document Type	File Types*
Word Processing	doc, docx, docm, rtf, txt, xml, odt, wps, wpd
Spreadsheets	xls, xlsx, xlsm, xlsb, xml, csv, dif, ods
Presentations	ppt, pptx, pptm, xml, odp, pps, ppsx, ppsm
InfoPath	xml, infopathxml
Publisher	pub (2000-2010)
Email	msg, eml
Visio & Vector formats	vsd, vdx, svg, svgz
HTML & Images	html, mht, aspx, jsp, gif, jpg, bmp, png, tiff
AutoCAD 2D / 3D	dwg, dxf

In addition to converting to PDF, it can also be used to cross-convert between document types (*Word* to *Excel, Excel* to *Word, DOC* to *DOCX*), watermark, split, merge, and secure documents– all from SharePoint Designer and Nintex Workflows, a friendly SharePoint User Interface, and from your own code with a professional, stable, and fully documented object model.

Operating Systems	Windows Server 2003-32 / 64 bit inc R2 Windows Server 2008-32 / 64 bit inc R2 Windows Server 2012
SharePoint versions	WSS3, MOSS2007 SharePoint Foundation 2010 / SharePoint Server 2010 SharePoint Foundation 2013 / SharePoint Server 2013
Browser versions	Internet Explorer 6, 7, 8, 9, 10 Firefox, Chrome
Memory	Depends on the size and complexity of the documents that are converted and the number of concurrent conversions taking place. We recommend a minimum of 1.5GB of total memory.
CPU	Any CPU that can comfortably run SharePoint will be suitable. We recommend one or more multi-core CPUs.
Disk Space	This product requires 20MB of disk space.

Organizations relying on InfoPath to store important data may not realize that opening these forms in the future will be problematic due to the typical entropy encountered in IT environments.

If this is the case in your organization then accept the inevitable and **ACT NOW** by converting InfoPath forms, including all attachments and metadata, to a more durable format such as PDF or MS-Word.

The Muhimbi PDF Converter for SharePoint makes it easy to automate the conversion of various file formats, including all attachments, to file formats more suitable to archiving such as PDF, PDF/A1b, PDF/A2b, MS-Word, Excel and HTML.

PROBLEM: Sharing InfoPath forms with third parties is nearly impossible.

PROBLEM: Opening existing InfoPath forms 3, 5, 10 years from now is unlikely to work.

SOLUTION: Convert your forms to PDF, MS-Word or Excel to safeguard them for the future.

This whitepaper focuses on archiving InfoPath Forms, but the same can be said for archiving HTML pages or MSG based emails, file formats all supported by the Muhimbi PDF Converter for SharePoint.

muhimbi

Contact us at **support@muhimbi.com** for details or to book a demonstration.