Lukas (00:00):

Alright, so thanks for joining everybody. So today we're going to talk about Typefi automated publishing software and how that software can help you automate financial and insurance documents. So we've got three people from Typefi on the call today. Caleb Clauset, he's our VP of Product. He's going to do the demonstrations. Then we have Eric Damitz, who's a Senior Solutions Consultant. He'll be around for answering questions. And then I am Lukas Kaefer, I'm the Marketing Manager. So quick recap, if you're not familiar with Zoom, you have mute buttons on the left side at the bottom, chat button somewhere in the middle, and then a raise hand button as well. We ask that you do stay muted during the presentation, but if you have a question, please post it in the chat or save it to the end. Please feel free to unmute during the Q&A. There will be a Q&A at the end of the presentation and then we'll send out the recording within a day.

(01:07):

Quick background about Typefi. The company was founded in 2001 and is based in Australia. Our main product is an automated publishing software for Adobe InDesign and that's what we will be showing you today. We also have a few plugins for InDesign and a service called RunScript that is basically cloud services for InDesign Server. And I'll post some links to those in the chat in just a minute. But this is the quick, this is what Typefi actually does. Maybe you've seen this before. So we'll explain a little bit about what's actually happening here and how this actually works today in the demo. But basically the real value of Typefi is that it can help you publish faster and it can help you do more. So you can publish up to a thousand pages per hour in InDesign. You can connect it to your CMS or your database with an API and you can even publish accessible content using that same source content used for other outputs. And there's a quote from one of our clients, Rhino-Rack. They actually cut their production time for all their product guides and manuals from one month to just three days. So massive time savings. And that's really it for my intro, so I'm going to go ahead and turn it over to Caleb.

Caleb (02:33):

Alright, just a quick introduction for myself. I'm Caleb Clauset, I'm VP Product. I've been with Typefi now for just shy of 20 years and it's given me opportunity to work with many different types of customers and see just how unique the challenges are, and yet the solution that we have is broad enough and flexible enough to address many of those challenges in a time that can be a game changer for many. So what I wanted to start with is just looking at an example of a workflow that is publishing content and this particular workflow, what we're looking at on my screen right now is a very simple workflow with four stepsa as part of that. We start with a Microsoft Word file as our source content. This is being ingested and transformed into an intermediate XML file and then we're automating composition. So we have to gather the fonts we're using for this publication so that we can be brand compliant and then we're creating an InDesign file from that and then exporting that out into a desired PDF standard.

(04:03):

So when I run this particular job, this will trigger, through the APIs, a call to our server and a call to InDesign Server to compose this document. Now this is a fund fact sheet. You can see it's already complete, it's done. So we're going to open up the PDF and we can see the way that it has rendered all of this content. We have charts and tables front and back. It's all brand compliant. One of the key things that we see in the financial services industries are regulatory requirements, that you have to have things a certain font size, you have to have standards in the way that you present information. You've got lots of boilerplate content and disclaimers that are part of this, and all of this can be automated as part of this composition so that you have absolute trust and confidence in the fidelity of what is composed using the solution. That there's no copy paste or possibility for errors to be introduced by a person. Now this particular design, things like this table here on the back, this can actually be rendered on the fly from content that's pulled through say an SQL query or so forth, raw data coming into the system, and then exporting that into the particular style and appearance that we want to have for this content.

(05:46):

So when I look back at the InDesign document, let me grab that. All right, so now we can see this loaded up into InDesign and just the things I wanted to point out in this content is that there is structure that's been applied to the content. That we know this is a chart or graph, they're greyed out because I don't have the actual links to the art on my computer, but we have this other thing called a sidebar. So we know how to route content into specific regions within your content. And so all of this is very easy to set up from a design perspective and all of the intelligence and the rules about how things are laid out is driven by this design. So it makes it very easy to adapt this into other formats. So let's switch back over into a more advanced demo. Oh, sorry, I know where I am. I need to switch up, there.

(07:03):

So as an example of how we can integrate with, on the API side of things, this is using Adobe Experience Manager as a front end authoring tool. And so here I'm within the Assets view and I have a whole series of component pieces that are set up within Experience Manager. I'm actually using AEM Guides as my authoring component, so these are all DITA files and I can edit individual DITA files. This is my DITA map that shows, here's what my publication should be composed of, with the different topic groups and the references pointing to all these discreet pieces of content. This is really important when we start thinking about content reuse. And again in this notion of standardisation with boilerplate content, that all of these things are going to reference some of the similar pieces. And so I want to author it once and use it in multiple locations.

(08:13):

And so this is a solution where we have built a custom integration within Experience Manager so that I can literally select my DITA map and because of the Typefi add-on, we can see that we have a new link to go into our composition engine. So I can choose Typefi, I can choose what kind of output I want to return back from the Typefi engine. I can choose different workflows. Typefi also supports pretty much all the world's languages. And so if you're doing English content or Chinese content, it's the same system behind the scenes. It's just a matter of making sure you have the right fonts and that your content has been translated through a service prior to Typefi. So I'm going to choose that. I want to get everything back into AEM and I can run that. And this again is using the Typefi Server API to send over all of the content from within Adobe Experience Manager Guides and Assets to the Typefi Server.

(09:28):

We can watch that, sort of tail the log, but once this is done, this takes about a minute and 20 seconds I believe, that this content is then returned back to AEM so that it can then flow into the next stage in your workflow. So if you're going to then move this into a content review cycle that, before it's published to your website or distributed in that fashion, that can happen there. I don't have the demo right now, but we've also done integrations with Adobe Workfront as a workflow management tool. And so the same sort of process follows there where you can trigger a job from within a Workfront task or step that will then render the output according to the rules that you've set as a designer and the rules that you've set as your content, and then return that output right back into Workfront for whatever happens next.

(10:37):

So we'll download the output from AEM, and so here's the full document that's been rendered from that DITA source. This runs about 24 pages. You can see in the front we had some multilingual content. All your hyperlinks are coming through as hyperlinks. We get into our various T tables for the content, to describe how to resolve issues. All of the formatting, the structuring, and the types of content you have from heading levels and so forth are passed through. What's harder to sort of demo is the fact that all of this content is accessible, meaning that the Typefi engine automatically handles the structuring and the tagging and the markup of that content based on the semantic source content files. You have to pass that through so that your PDFs that we generate are fully Section 508 compliant, and that is the super fast walkthrough of an end-to-end solution using Typefi to automate composition of different kinds of content using APIs or directly from within our workflows to render out with content velocity, as Adobe likes to say, that the speed at which you can render these things is far faster than anyone could achieve this through the traditional desktop publishing workflows.

(12:34):

I know we have several people on here. Do we have, I've not been looking at anything in the chat. Do we have any questions that have come up or there's anything that people would like to see in more detail from what we've done so far?

Lukas (13:00):

Yeah, I don't see any questions in the chat, but yeah, feel free, if anybody has any questions, unmute. Oh, we just got one. Can you see that Caleb from Sylvia?

Caleb (13:16):

Yeah, what are the source files you start with? Okay, so the source files from a Typefi perspective, it can be any kind of XML. We have in this particular example, this is using DITA XML, which is really commonly seen in manufacturing or technical documentation, but we also support other kinds of XML, whether it's, we work with organisations like the ISO and various standards bodies, and so they're using the NISO STS XML. We work with organisations like the World Health Organisation and the New England Journal of Medicine, and they might be using JATS or NLM XML. Other publishers are using BITS XML. We've done DocBook and even proprietary XML standards that you might have in place. And obviously we also support Microsoft Word and Excel as data sources. So it just sort of runs the gamut as far as what you want to achieve or what kind of content you start with, but that's one of the things that will work with you to understand the rules that go behind that.

(14:45):

The second question was colour contrast. That is a design guideline within your actual, if you're using Typefi for your output, then that's part of what you would manage in your InDesign template. And so whatever you set up as the visual rules within InDesign, that's just going to be applied rigorously through Typefi. And so you're responsible for setting up that colour contrast in the colours that you choose for text and so forth. As far as do you need InDesign templates, one of the things that Typefi is actually quite good at is being able to take an existing design, whether it is in InDesign or whether it's a PDF, and then creating an automateable version of that as InDesign. We do use InDesign as our rendition engine, and so all of this content that is going out into PDF is composed through InDesign, or InDesign Server, technically.

(15:53):

And so yes, there are InDesign templates behind the scenes. So when we look at this workflow that was running the content from AEM, this is a similar sort of four stage workflow. First stage is we're ingesting the source content. This is an AEM package of content coming from there. Then we gather those fonts again so that we can stay brand compliant and compliant with legal. And then we have the InDesign document task where we're pulling in an InDesign template as the base set of rules and we can augment that template and extend it through what we call event scripts. And so when we have really complex layout requirements, we might have other things that we want to do within that event scripting. So here we have a special rule around how hyperlinks that run long should break within a column. And so this is implementing those rules for you, and I think that should sort of answer that question.

(17:00):

If there's any follow-up for that, I'm happy to go on. The depth of integration with AEM. So Typefi is pulling data. So from within AEM, for the example that I did earlier with Guides, we're actually using, our integration is built around the Guides add-on for AEM. And so we're using Guides APIs to parse the DITA map and locate all of the different DITA topics that are referenced within that DITA map, package all of that together into a zip package, and send that over to Typefi. The same logic can also apply when you're dealing with content fragments that might be gathered together into an experience. And so that experience is pointing to many different content fragments. We would bundle those pieces together and pass them through our API to transform and output into InDesign. So all of that is possible. If you have other CMSs that are not AEM, the same logic applies. We have implementations that are using Alfresco, Documentum, Mark Logic and others, even SharePoint as a CMS to push content through Typefi and return, always, whatever the outputs from Typefi are back into that CMS for the next stage so that it's not a one way path, but Typefi is a step in your overall workflow.

(19:05):

One more on AEM. Does this apply to AEM server and or Cloud? So yeah, that's a great question. So what we're showing right now, this is an on-prem installation of Adobe Experience Manager and the key difference between the AEM on-prem versus the AEM cloud is that the management or configuration of the add-on for AEM cloud is, for security reasons, much more distant. And that's an aspect of our current connector that is dependent right now on the on-prem deployment, but that's an active work on our side to just add that additional abstraction layer to fully support AEM in the cloud. There's a third choice, which is the AEM Managed Services version, but that's just an AEM on-prem that Adobe is managing for you.

(20:10):

Any other questions? Was there a specific question about SharePoint or just the fact that we're using SharePoint or we have customers that are using SharePoint to drive workflows? I guess I should point out that from a content perspective that all the documentation and specifications around how you can integrate Typefi are just available through our help site: help.typefi.com. Under development resources here we can get into understanding what the API from the Typefi Server is, as well as understanding the structure and the schema of that intermediate XML that we're transforming to. So we're going to take whatever the source XML is, whether it's DITA or Office Open XML or NISO STS, that's transformed into Content XML, which is optimised for that output and publishing with InDesign Server. I'm also wondering what all the outputs were. Well, from an output perspective, this is one of the things to think about within the Typefi side of things is that the workflows that we were showing before.

(21:42):

So when we look at this Create PDF workflow, it's got four steps: the import DOCX, the gathering the fonts, create InDesign document, and export PDF. But the options that you have within Typefi are quite extensive. We have a number of actions, discreet actions that are available for InDesign itself. We integrate with ANT to do more complex command line type workings. DAISY is a pretty esoteric file format that we export to that this will drive a braille device, which is kind of cool. We've got the Import DITA we were using before. We integrate with Delta XML to do diff-compare. And so if you want to compare what's changed between different iterations of a publication, Delta XML is absolutely the way to go for that. HTML export, the support for Markdown. Round tripping with Microsoft Word is yet another option. This gives you some flexibility to say that I want to render my content into InDesign, but then I want to be able to take any changes I make in that InDesign file and export them back out into a structured Microsoft Word file.

(23:12):

So that's another action or output that Typefi supports. We also integrate with PDF Box as an add-on to manipulate or post-process the PDFs that are generated from InDesign Server for cases when there are additional tweaks or settings that you would like to make to those PDFs that are not natively supported with the commands that are available from within InDesign. Now this last one down here for Run Script, this is probably the most powerful action that Typefi supports, and this allows you to structure your own extensions using Windows host scripting or any other scripting language to drive and manipulate the content. And so if you want to create your own custom action without having to use our APIs, but just, I want to run a script that does X, Y, and Z, then you can drive that with our Run Script interface. There are a lot of security implications with this and this is why it's sort of more controlled and has to be managed in a more tightly sort of gatekeep behaviour.

(24:33):

But yes, we do offer even greater extensibility in that sense. Here's the two different copy to and, copy from and write to SharePoint as well as various utility tags and integration with validations to validate URLs and schematron and so forth. And that's a rundown of the different actions that you can add within your workflows, and these actions are super easy to integrate with and extend. So I can take a workflow like this and just add an action, choose from the list. So if I want to, in addition to making that PDF, I also want to export this into an EPUB, I just add that action to the end and I can choose, it automatically fills in different steps, and make my different choices here. And then the next time I run this job, it will produce an EPUB as well.

(25:40):

All right. Interesting question from Ryan about PowerPoint. I know we don't have much time left, but the crazy thing that we actually offer now is that you can go from structured content in InDesign to a structured PowerPoint presentation. So this is a unique offering that Typefi is now able to deliver of being able to write out a PowerPoint deck from your source InDesign file. So as far as importing the PowerPoints, that's not something that we've done and primarily because there's very little structure, or semantic structure, in the way that content is authored within PowerPoint to be able to glean out and figure out how things need to be arranged and semantically structured within the XML to drive that. But the PowerPoint file that we produce from InDesign is a well-structured, well layered document, it's pixel accurate to your rendition and uses all the same fonts and so forth that you might have referenced within that layout. As far as accessibility, we do the same sort of things we do with PDF and EPUB so that if you have, within your source content you've provided a way for us to capture or to look up the alt text, then we will apply that through and through so that when we generate out, that your content is as accessible as possible.

(27:39):

Alright, let me think. Lemme go back through the questions. Yeah. Okay, I think those are all the questions. I think one of the things, Eric, would you mind talking a bit about the way that we did a project, it was all around the templates. This was for, the name escapes me right now, Lockton.

Eric (28:20):

Sure. Lockton is an insurance broker and their issue was every year they publish a new proposal for all their clients and these proposals need to be brand compliant, they need to look nice. They have a tonne of account managers who are sending all this information to their design team and saying, I need these PDFs tomorrow. So their design team is being swamped by all these requests coming at random times and they need to create all these perfect InDesign documents to make these PDFs. And some of them need to be landscapes, some of them need to be portrait. So they have a lot of different design requirements. They were working in Word, and we went and talked to them and kind of looked at their editorial process and on their end they decided Word wasn't the right way to manage all this information. A lot of it is boilerplate stuff.

(29:16):

Some of it is repeated between documents, some of it is new for a specific customer and a lot of it is tables with pricing information and things in it. So what they decided to do was develop a new front end, a web-based front end, so that their account managers could go in there themselves and just enter the information they wanted to. They could upload Excel files with all that pricing information and then all of that information would be managed through their web front end and their content management system so that when they went to publish, what was coming out of the content management system was that Typefi CXML Caleb mentioned. So everything was set up to be able to be published through InDesign via Typefi. On the Typefi end, all we really needed to do was help them create a couple InDesign templates, the portrait and the landscape version of it, enter some boilerplate content into the template, which wasn't coming from the account managers, it was just actually in the template itself.

(30:20):

So when that page would appear in the document, everybody got the same page basically because that's what was in the template. And the way the system works now is their account managers can go in, set up a little project, put in the title, all the information they want, they can upload their Excel files with the information, and then when they hit publish, they get their PDF and they also get an InDesign file. So sometimes they put all their information in, they get a 95% complete document that they like, but then they can also give that InDesign file to their design team who now, instead of having to do the entire thing from scratch, can make the couple tweaks that particular account manager wants made for that particular file. So I can't remember what their time savings was, but it was pretty startling. I think they were even surprised by how big that change was.

(31:15):

So now they can publish these things much more quickly, fewer or very few errors because they're not copying and pasting and passing files around between different groups of people. And honestly, I hear from them maybe once a year, every once in a while they have to update the boilerplate. They win some other award, so they want to put that on the back cover or something like that. But it's a pretty smooth, once everything gets up and running, it's a pretty smooth process because it's kind of doing the same thing over and over. It's just different numbers, different content, but it's the same pages. They look the same. Every one of their clients gets something that looks similar. It's a good way to keep that branding. Actually, they did a branding change kind of a couple of years ago and all we needed to do was update the InDesign templates. So they wanted to add a black bar to the top of a page, so add that to the InDesign template. Now every file published from that from then on has the new black bar at the top of the page. So once the system is set up and running, it's really easy to make changes, design type changes like that. So it was a really successful little project that we did with them.

Caleb (32:25):

Awesome. Thank you Eric. I know we're actually just over our time for this time for the webinar. I want to thank you all for joining us. Thank you for the questions. It's always, I enjoy the interaction. If you have any follow-up questions or things, Lukas has just posted into the chat our contact information and some links, we would love to hear from you and see what we can do to help you do more. If there's nothing else, thank you for your time and we will hopefully chat with you next time. Thanks so much.

Lukas (33:02):

Thanks everybody.