

Arjeh ([00:01](#)):

Hopefully everybody can see our screen now.

Audience ([00:05](#)):

Yep. All good? Yep.

Arjeh ([00:08](#)):

Perfect. I'm also audible?

([00:17](#)):

Yep.

([00:17](#)):

Okay, great. Then let's get started. So thanks for the introduction and for having us speak today, Chandi and Lukas. Thanks for coordinating us being available for today to present some case studies. Hello everyone. Even although I partially introduced myself already in one of the breakout sessions today, my colleague Nalaka and I will talk you through some use cases of Typefi implementations we did for our customers. But when I say customers, what does that actually mean? Let me give you a short introduction of what Innodata is, who we are. This will take just five minutes. It's more important that Nalaka talks because he will dive into the use cases and a short demo which will take around 15 to 20 minutes. Now, should you have any questions along the way, we encourage you to please note them because at the end of our talk we will reserve some time for questions and answers.

([01:12](#)):

So I'm Arjeh. I'm based out of the Middle East. I'm at Innodata for 10 years now and have a long history of working in the publishing and information providing business. And among others I support our publishing and information providing customers with implementing and optimising their operational publishing workflows and processes. Now my colleague Nalaka Fernando, based out of Colombo, Sri Lanka, is responsible for the successful implementation of our technological solutions with these clients and he is also working on Typefi technology in our platform. And Nalaka with the help of Deepak and Sameera is going to be demoing our use cases that we prepared for today's session. So very briefly, some words on Innodata. We are a global data engineering company that help leading enterprises in different market verticals to drive operational excellence by the use of our AI driven solutions, platforms, and subject matter experts. Our head office is based in New Jersey. We're traded on the NASDAQ Stock Exchange and last year we celebrated our 30th birthday. Our company has offices and facilities all over the globe from the US and Canada through Europe, the Middle East and Asia. And we're just over, I think today 4,000 subject matter experts and technology subject specialists.

([02:46](#)):

Our client base is quite diverse as we work, for instance, the top five technology companies in the world, supporting them with AI services to help them further develop their foundational large language models. I think Caleb already talked a little bit about Gen AI. This is also of course for us an important direction where we see that our customers are really developing capabilities but also are looking for implementation possibilities. And especially our corporate product and services providing customers, our information providers and publisher customers, are part of our customer base who are also utilising our services. Now we do these services for clients across a vast number of domains. We have customers in the legal space, scientific, educational, medical and mobility sectors and we provide them with services in dozens of different languages. Now Innodata's added value for our customers is to provide economic benefits, of course, workflow efficiencies and time to market improvements. And in the publishing workflows we offer our clients our proprietary cloud-based, AI driven, intelligent automated publishing platform. It's a mouthful, so that's why abbreviated it to our platform called IAPP. Now the principles of

IAPP are actually quite simple. It's a management and production solution for end-to-end XML-based workflows which allow for authoring, editing, typesetting proofing, publishing and distribution of structured online, digital and print publications all from a single source of truth. I think Caleb also spoke a little bit about that.

(04:37):

Now our cloud platform contains modules that handle content creation, content editing, content management and performance tracking and monitoring. And one of its driving components is the Typefi intelligent publishing automation software. Now typically an optimised end-to-end publishing workflow for us looks like something we portray on this page. We're a bit short on time so we will not go into all these workflow steps now, but we provide our customers more or less from the creation of content publications all the way to the downstream distribution of that. But Nalaka will address some of these steps in the workflow as he goes through the demo and the use cases that he's going to present to you all today. That was my part of the introduction about Innodata and how we support our customers. Nalaka, may I ask you to take over from here?

Nalaka (05:39):

Thanks Arjeh. Yeah, so I think you all got a overview about this end-to-end publishing platform. So I'll just give a little little detail and then we'll jump into a demonstration and then we'll discuss a few case studies how Typefi helped us to implement this platform. So if you look at it like this publishing platform, so we call it as IAPP. So this has main four modules where the publication planning can happen. So where the publication management team can do the publication planning and allocation to the authors. So the authors will work on the content, they write the content or either they edit the content. And also we do have option where we can absorb the Word files as well if the authors have written the content outside of the platform. And the advantage of this platform is, this platform is modularized. So where we have connected the content harmonisation module. So where if authors writing the content outside of the platform, so when they drop the Word file, so there's an AI based module which identifies the content structures and transforms it into structured content and then it'll be available for the further editing in the publishing platform in XML format.

(07:07):

So once the editing is completed, so then it goes to the typesetting flow, so that is backed by Typefi. So we are using several features of Typefi in order to automate the publishing process. So this process is automated, so at the click of a button, it goes to Typefi and then it creates all the deliverables like PDF and Word outputs, EPUB, and the XML also. So that is through Typefi we were able to generate these automatically. And once it's generated, it goes to the authors for their review. So this is the lifecycle, this is the cycle of this publishing platform. You do the planning, allocation, and then authors write the content or edit the content, and then it goes for the publishing platform where Typefi supports it to generate the PDF or the other outputs and go for the review. So we'll jump into a demonstration of this platform quickly and then we will touch base on how Typefi helped us to overcome certain challenges we had in the typesetting. Thanks Arjeh. So I want Sameera to share your screen and then go to the demonstration.

Sameera (08:25):

Yeah, thank you Nalaka. So I'm sharing my screen. Yeah, I hope all of you can see.

Nalaka (08:41):

Yes.

Sameera (08:41):

Yep. Okay. Alright. So what you can see in my screen is the IAPP platform. So I log into the platform as a sort of publication management level user and the first thing we can see is sort of a dashboard to track the different stages. And then there's a publication planning module where we can preplan our content lifecycle. So the scenario like we are going to explore here today is like if the authors are writing the content outside or maybe they can write the content directly within the platform and how they can generate the different kinds of layouts according to the project requirement through Typefi. So the first scenario, like Nalaka mentioned, if the authors are writing the content outside the platform, there's a module we can drop the Word content into the platform where it'll go through this Gen AI capabilities to automatically structure the content. So the segment that I'm going to demonstrate is, once this content gets transformed, how we can do some content editing and then generate the PDF output.

(10:03):

So this is one of the technical manuals from one of the customers. So this interface is a XML-based editing interface, but it'll present to the author as a Word like interface because most of the authors are not much familiar to the XML-level editing. So this is the real XML content but it gives sort of generic features as you've seen in Microsoft Word, like the inline styles, listing, tables, footnote, likewise many options. So in the left hand side it's showing the document structure and the right hand segment you have this XML level attribute information because still we are working on the XML. So I'll do a few edits and then let's generate the output from Typefi and let's see how it's going to look like. So I'll do a few content edits, content, and I'll delete some text. So these are getting track change.

(11:24):

I'll apply some inline styles, making the content bold, and I'll add one more list item, new list item. Then we'll also include a simple kind of a table, and some other thing we can input a small image as well. So I'll go to the image library and I'll put a simple image into the document. So I did a few quick edits into the document. Now what I'm going to do is I'm going to generate the PDF with the help of Typefi. So for that we do have a print option and we can generate the PDF.

(12:32):

So it'll take a few seconds based on the content length of the file depending on the number of pages we have. So in the background what is happening is we are transforming this XML into the CXML conversion to make it compatible with Typefi and then it'll ingest into the Typefi platform through the APIs. And once that PDF is generated through Typefi, we will be automatically loading into this interface. I think it get through. So this is kind of a simple and clean layout. So we created based on the customer expectation for the technical manuals, you can see the updates added to the content are reflecting over there including the image. So all these are coming through the template, how we defined it. So I'll show you another two examples that are from the different layouts, how we can customise these layouts from Typefi and get a different sort of output into this interface. And also we do have integrated this Word generation as well if authors need to download a copy of this document as a Word file. So this is also coming through Typefi, we can download the same content in the Word format. So I'll shift to another document.

Nalaka (14:19):

So that actually, so the advantage that we got from Typefi is to generate the Word files as well because we're having a separate template for that. And then we're generating Word outputs, so which we'll talk about in a while, like coming our Typefi expert from Innodata side, Deepak. So he will sort of touch base on a couple of case studies how we are using heavily Typefi features.

Sameera (14:46):

So this document is sort of medical domain content and as you can see the layout is different. It's a two column layout with a lot of colours and the background information. So this template we are generating

from Typefi based on the customer requirement. And the next one I'm going to show from the sort of legal content.

Nalaka ([15:16](#)):

So these formats are autogenerated using the Typefi features and also doing several scripting from our side as well. But it's supported heavily by Typefi features to get these automated.

Sameera ([15:32](#)):

So this is a legal content, if you look at there are some specific things like marginal numbers and the footnotes. So these are generated from Typefi. And different kind of box layouts to preserve some of the primary content. And then if I go a little bit down I can show you a few table structures as well. Okay. There's a two column table with one structure and if I further scroll down there's another table example. So in the legal content, I think one friction like Nalaka initially raised as well, we are having a couple of hundred page length document as well. Table is continuing to maybe sometimes it's more than 200, 300 pages even, but still we have managed to generate those kind of tables from Typefi. So that's a plus point. So all these, to summarise again, we are generating through the templates with help of Typefi in a realtime interface from the platform.

Nalaka ([16:49](#)):

So what we do is, I mean as I mentioned earlier, so once these are generated automatically, so then it goes to the authors for their review. So the advantage of this platform is, authors can do their author updates, comments, reviews, directly into the platform with comments and with track changes so that it'll be reflected in the content and then it can generate the PDF, the authors, editors can generate the PDF and they can see it real time with these features. I think Sameera, that's all from your side?

Sameera ([17:27](#)):

Yeah, that's what I have had to showcase today.

Nalaka ([17:30](#)):

Great, good. So let's move into the slide deck again Sameera, and we'll touch base on a couple of case studies. What are the features of Typefi we used and then how those features helped us. You can go to the next slide.

([17:51](#)):

Right.

([17:56](#)):

Yeah, so I would like us, Deepak, to talk a little about and then I will also. Deepak is the Typefi Specialist, actually he got trained from Typefi team. And we have a set of team that got trained from Typefi team. And Deepak, you want to explain a little bit of these case studies?

Deepak ([18:19](#)):

Yeah, so hi, I'm Deepak. So I'm working as a Typefi template expert here. So the few case studies you can see on the screen, so as Typefi supports, like Typefi provides a different workflow and actions for different kinds of layout creation. For instance, you can see in the left screenshot where one of our customer requirements was that they have one template but within the one template, based on some XML attributes, they want to change the layout on the fly. So as Typefi supports the Java scripting at different levels of events, we were able to achieve this layout with the one template. You can see on the screen

there are 14 kinds of layout requirements of the customer. And based on this 1111, 1211, we have to read the XML and manipulate the layout on the fly. And with Typefi and the support of the JavaScript from different levels like document start and document end, page end, spill, we were able to achieve that. That was a great advantage of Typefi and the support, which the way Typefi supports the Java scripting was very good for us and rather than creating different type of templates, 14 templates, we were able to achieve in a single template all 14 types of layouts.

(19:46):

And apart from that, as there are some limitations within InDesign where it doesn't support the complex tables, it doesn't support some background behind the tables, and there's some footnotes issues also there but with the help of Typefi and the scripting support we were able to overcome. Earlier, even though we have Typefi, but because of the InDesign limitation we have to manage those manually. But with the support of this JavaScript at the backend, we were able to achieve those layouts. Can you move to the next slide?

Sameera (20:24):

Yeah.

Deepak (20:28):

And because Typefi has a plug-in within InDesign. So with InDesign and Typefi, we can manage any complex layout, any complex scenario. As you can see in the screen, we have floats within the body, we have floats through. So Typefi gives us more flexibility to define the pagination rules within InDesign with the help of Typefi plug-ins. So we can decide, okay, which table has to be present where, if we need to. So the way they are supporting the tables with the cell style, which is not quite easy with the traditional ways or manual. So the way they are supporting, you can see in the screenshot where in one table we have different shades of the colour. So with Typefi we can read the XML and we can manipulate the layout or we can define the style sheet in the template.

(21:24):

Similarly, you can see different types of pagination rules we can manage, we can define in the template where my float should appear, how it should look like, and what kind of colour. Even in one project, a requirement was that in each part, the colour has to be changed. So we define all the colours in the template and through Typefi and the JavaScript on the fly, we change the colour by reading XML. Can you move to the next? So you can see on the next slide also we have some more layout differences and with the Typefi InDesign template and Typefi pagination rule, we were able to achieve that. Can you move next?

(22:16):

And so this is one more feature we have in Typefi where not even the template and the plug-in and the script, it allows us to read their Content XML within the template, which is a very good feature. In one of our projects, the problem was that while Typefi supports the index options, a requirement was in this project, which you can see in the screen, that they have the entire index XML, but they wanted to add the page numbers at the end. So through scripting, Typefi allowed us to read their Content XML. And with these IDs you can see in the left screenshot, we were able to get, with these IDs, we were able to get all the page numbers in the right screenshot onto the proper index entry. So with the normal traditional way, it's like all manual, somebody has to check those entries and see where the entry on the page is and needs to update because here they have given entire XML with the ID. So with the help of JavaScript within Typefi, we read their CXML content and we're able to get the exact page location and update the CXML and generate output.

(23:37):

And one more thing I want to add here. Earlier before Typefi, because we have so much legal content where the book volumes are very high, sometime it goes a thousand, more than a thousand pages, 5,000 pages. In the earlier process, the team had to do those things manually. It was a very tedious task. They were taking a lot of time. And with that Typefi support, we can just push the XML within Typefi and define the templates. And now in a few hours we were able to generate thousands of pages from the backend. And during that time, a user can do other work and so we save a lot of time in the composition especially. Yeah. Thank you. This is for my side.

Nalaka ([24:28](#)):

And also one thing, a few things to touch base, because earlier when we didn't have the Typefi solution. So we were doing this typesetting work manually. So that actually, compositor, typesetter, was able to do 30 to 40 pages per day. But with Typefi we were able to increase drastically, around three, 400 pages. So now a typesetter can do it with all these automations, so it's like automated publishing. So with the support from Typefi, we were able to achieve that. That is something I want to highlight here. And also that actually helped us to achieve the turnaround times because earlier the turnaround times spread across like weeks, but we were able to deliver these turnaround times for customers really quickly. I mean within days we were able to achieve that.