Lukas (00:00):

Next up. Last but not least, we have got Caleb and he's going to talk about the future a little bit. So over to you Caleb. I think you're muted.

Caleb (00:19):

There we go. I had to find a little tiny icon. Okay, so using this as the basis for doing more in Typefi and how that looks going forward. First off, just looking at this stack of how all of our different products fall into one another. The top row, we have our various clients: InDesign, Word, with our Designer and AutoFit tools, Typefitter, Writer using MathType for creating equations, various web clients. Then that middle layer where we have the traditional Typefi Server 8 sort of workflows, which has its own file management. We also now integrate with Fonto and with the Oxygen Web Author. So you can actually from within Typefi Server 8, you can edit and manipulate your XML files using either one of those editors. Workflow plug-ins, AssetFlow, AEM Guides, all as this sort of content management layer. Blinkenlights in the backend. This is how we distribute all of your jobs across a farm, I guess, of InDesign Server instances. If you're on-prem, you can also have Blinkenlights to handle multiple InDesign Server instances to distribute those jobs across the board so that you get more output in parallel.

(02:10):

And then just in some ways, this is kind of a scary diagram to me just looking at all the different technology stacks that we are using, but the reason why I call this out is really just trying to say, we're trying to position ourselves in the best possible space for building these tools for you. So we're using C# to build the Writer plug-in within Microsoft Word because that's the native architecture for those add-ins. We're using C++ for InDesign because that gives us the most full featured APIs to create interfaces to create the tools to manipulate and set up those templates. And so this is just sort of a diagram looking across all the different technologies and the technologies that are behind each of our components.

(03:16):

Now this, this is sort of this obligatory "lawyer made me say this" big scary slide about roadmaps, that these are just thoughts and that features and functionality that I'm talking about today may or may not happen depending on when things happen.

(03:38):

So the major themes as far as where Typefi is interested in and thinking about and sort of planning around this doing more. So we talked about the Typefi Cloud 3.0, which is sort of happening now. This is that foundational work of moving to Java 21. This is the Tomcat 10. This is the Auth0 for single sign on. The launch of AssetFlow and that rollout over this year. It's a big sort of revamp. We launched Typefi 8 back in 2015. So this is part of that cycle of renewal. And so this is a transition time for us. We're also really interested in collaboration and thinking about what it means to collaborate and how collaboration starts to impact the core automated composition that Typefi has been known for in the last 20 years. And I've insinuated some of this in the breakout sessions, but Gen AI is also a big part of how we think about this model.

(04:57):

Generative AI and machine learning in general are great at dealing with quantities of content that is just beyond the human scale. How do we pull that into Typefi? So here's the recap, what we talked about in my first session around all these different things. Collaboration—what are we talking about with collaboration? So one of the first things that we have been sort of tinkering with, and to be honest, this idea of the Advanced Engine and recompose, I think the first time that we internally within Typefi talked about this was probably 2008. And thinking, what does this look like? How can we start thinking about this idea that when Typefi delivers an output that it's not sort of locked in and the next time you run the job you get a brand new file and all the decisions are refigured all over again.

(06:09):

How can we say, oh, I want to lock part of the file down, but this other content I want to swap out. I want to replace a story here. I want to replace a table there. I want to replace some pricing information there. I need to swap out the artwork because I had dummy content before, now I have the actual licenced artwork that goes in. But I don't want to change anything from all the decisions that were made either by Typefi's automation or in a post-process automation where you have picked up that InDesign file that Typefi created, and then you've moved things around and made your own decisions—that we could still work with that file and push updated content into that. That's what the Advanced Rngine is all about. Project Skywriter, this is something that I gave to Guy about a year and a bit ago, and the general idea behind this is Microsoft Word is a very different beast compared to whether it's on your desktop versus in your browser or on a tablet. And the way that Typefi Writer functions, we can't do the things that we want to do either in the browser or on the tablet. So we're sort of confined to your desktop. How do we bring that out and unleash it from that particular device so that you can work from anywhere with any device. And so this is something that we've been working on in a skywriter sort of model and hope to have something to actually start showing and getting some feedback on towards the end of this year, early next year.

(08:05):

Asset workflows, this is taking on the AssetFlow tool, the front end I showed earlier and thinking about, okay, well we have these workflows that are built around composition and a stage based process. We want to take this XML, we're going to transform it, we're going to take that transformed content, push it through to InDesign, we're going to make a PDF from there, and then it's going to go somewhere else. But what about the workflows before that? What about the workflows after that? So thinking about workflows in that editorial cycle that we need to go through these different stages and the stages are associated with different groups maybe or different people post-process from Typefi. We want to go through a delivery process. Where does this go for approval? How does that figure into these different asset workflows? Not about composition, it's about this tracking and management and collaboration in that whole content life cycle.

(09:11):

And then co-everything is just like a catchall kitchen sink thing that we're still wanting to make sure that the way that we approach all of these things, that it's constantly seeking out your input and how you are using our tools with other stuff. We recognise that Typefi is not an island, that there are very few customers that start in Typefi, end in Typefi, and that's it. That there are always these other systems that integrate with Typefi before and afterwards and wanting to make sure that we are not doing things in our side of things that make it more challenging to integrate with these other processes. And so we want to make sure that we do the work that's necessary to improve that ability to integrate with those other things.

(10:10):

All right. Now, Gen AI, I mean, I think there's a lot of distrust that's kind of built in these early stages around Gen AI and sort of what it does. And there are certain things that Gen AI will do that are very impressive, but once you sort of dig into them, it's like, okay, this looks good on stage, but how does this work in an actual production workflow? And so the way that we are approaching this is very much in a hybrid approach. We want to balance those efficiencies of the lights out automation and the creative freedom that you have with making things more efficient and streamlining some aspects of that content creation. So Project Ragtime. Ragtime is a Retrieval Augmented Generation sort of tool. And this is an interesting thing and we're sort of looking at this initially as sort of an onboard version of an AI help assistant, like a trainer that, within each of our products, that we can feed as source data into this RAG system, all of the help content.

(11:44):

And so if you're trying to do something that you can just write a query to Project Ragtime and it will parse all of our help content and return back a result. Think back to the days when Google search results were actually useful as opposed to hallucinating things. This is about looking at the actual content and returning literal snippets of content or synthesising pieces of content from our source data so we know the quality of that data is good. It's not mixing in results from Reddit or some satire site. This is all about our site to help you understand how to do things within that tool. So that's where we're starting to think about how AI can augment your own abilities. Project Hemingway, this is something that we're thinking about around, I think Christina is the one that said something around the idea of, "Now if the Gen AI could say, this is a widow, this is an orphan and it could fix it for me,"—that's exactly the idea behind Project Hemingway.

([12:56](#)):

It is taking Typefitter as a tool and putting it into an autopilot mode to say, oh, hey, I found this thing. How can I tweak the text and maybe look at the text and say, I don't want you to write brand new text for me. I don't want you to even write a new sentence for me. I just want you to look at this text and say, can you squeeze out three characters somewhere to bring that widow back or bring that orphan back? You still have full complete control. Everything is going to be tagged as track changes. And so you can see before or after, you can reject or accept all those things. And that's sort of another way of thinking about how can we start to integrate natural language processing into this process of our tool, to be able to read your content and make those suggestions.

([13:52](#)):

Then you get to choose yes or no and how to work with them. Project Maestro is looking at tackling this challenge around images and thinking about how we can leverage and integrate Adobe Sensei machine learning into understanding and optimising images to be able to identify the subjects of a photograph of an image with on device-based cropping. So we actually don't change any of your data, but we look at, okay, here is the original piece of content that was provided as an image, in addition to thinking about the processing instructions that might say this is a small, medium or large image, that we can also look at say, well, if I crop it this way, then this actually allows more content to flow on that page and that's a more efficient, higher scoring overall page layout, but it's just a crop as opposed to slicing information out.

([14:56](#)):

And so again, you always have this backstop that you can go back to your original, you're not looking at something that's been created on the fly that's non repeatable. Project Titian is probably the most audacious concept that we were thinking around with Gen AI, is looking at these repetitive changes that you need to make across a single template or multiple templates. How can I basically have this AI assistant that I say, make this change, and it can go through your different open documents and say, I'm going to replicate that change across the board. Or I want to here create derivative paragraph styles based on this, but tweak it in this fashion. How do we create an alternative interface to the InDesign experience that is traditionally all mouse driven and something that we can automate that sort of processing. And so those are just four of the general ideas and how we're thinking around this idea of leveraging machine learning, leveraging the Generative AI in a way that augments and supplements the human designer, the human decisions, the human effort, to amplify your creativity.

([16:27](#)):

So again, this recap of where we're going with Typefi in the future. Right now it's this focus of delivering Typefi Cloud 3.0. Once we have delivered that initial launch, moving into what does collaboration look like? How can we increase the opportunities for collaboration and make Typefi more of a collaborative tool within these other tools and other systems that you're working in? Then lastly, how do we integrate and amplify your content automation using very targeted, very specific implementations of machine learning, natural language processing, RAG, Gen AI, across the board. And there's always this future of, what else? My personal philosophy on this is just about "yes, and..."—we want to do this and we want to

do more. We want to be a partner with you in this journey and the challenges that you face and the successes that you achieve. And that's where we get into this whole idea of we're not just doing, we're doing more. And that's what drives Typefi every day. That's my spiel. I don't know if we have, are we going to just do an open mic discussion now in the main room?

Lukas (18:03):
Yeah, I think so. If anybody has questions, go ahead.

Christina (18:14):
For the Project Ragtime you were talking about, was it Ragtime? No—Project Hemingway, you were talking about having the ability to take care of widows and orphans by tweaking the content. Would it be possible to have it instead tweak the paragraph settings instead of rewriting things? Okay, I'm just going to adjust the kerning just a hair instead of, because changing content for us got to go through people who understand finance and I don't have permission to change it because I don't really know what a derivative is. So even if it unless the basic typo and even then sometimes they get sent back to the SMEs, but if it was more like, okay, can we just adjust this setting so that it all nestles together or this table looks really crap, let's make it just go to the next page.

Caleb (19:10):
Yeah, yeah. So this is, Project Hemingway is sort of built on the foundation around what our Typefitter plug-in is. And so Typefitter has sort of two modes right now. So there's the interactive mode where you have a slider that you can drag tighter or looser to just select a paragraph and make it tighter, make it looser, and it will dynamically shift micro little settings and levers to move it and get things back and forth. You also have a fully automated fashion of you can create rules that apply there, but those rules have very narrow focuses. They're looking at a single page. And so part of, as a graphic designer, if I'm looking at content, widows and orphans, you might say, oh, well if I make a change to this paragraph two pages earlier and run this content a little bit longer, it fixes this problem two pages down the stream.

(20:16):
So we want to have sort of widening the blinders to look across my content. And there are other techniques that I remember from my design days of, oh, you're allowed to run this page short a line or this page long a line to balance things out. And so how do we start to integrate that sort of stuff into our workflows? That would also fit in with the Project Hemingway sort of umbrella. The last thing we've been looking at on that regard is especially in content where you have a mixture of texts and graphics, oftentimes you have, here's a hard and fast white space requirement around that graphic. Well, if I could cheat a little bit here and just take away a little bit of that margin from that graphic, that might allow my content to flow a little better. And so that's also part of this process where we have to use different techniques to say, oh, well, I'm now looking, I'm not looking at the text level. I'm looking at the page as an image and figuring out how to analyse that and where I can start to make those interventions. So yeah, we're looking at all of the above in this Project Hemingway, to give you that control of, do I want the AI to rewrite the content or to inject content or surgically remove content? Or do I just want it to squeeze it in a different way or make other changes to the geometry of the page to allow that content to fit?

Christina (22:06):
Thanks.

Caleb (22:08):

Anyone else? I think probably if you raise your hand or throw it in the chat. And if we have a real quiet bunch, then I think we will probably just call it earlier and give you guys back time for the rest of your day.

Lukas (22:29):

Sure. Yeah. Sounds good. All right. Thanks everybody.