Oral History of Lawrence A. Welke

Interviewed by:
Luanne Johnson

Recorded: April 7, 1986
Houston, Texas

CHM Reference number: X4597.2008

Table of Contents

Lawrence A. (Larry) Welke

Conducted by Luanne Johnson

Abstract: Larry Welke describes the founding of ICP (International Computer Programs, Inc.) to produce a subscriber-based catalog of software products in 1967, how he expanded into advertising-based publications in the early 1970s and how he established the Million Dollar Awards program which recognized successful software products. He covers the environment for software product vendors in the late 1960s and early 1970s including the challenges of finding the right pricing formulas for software products, the resistance of users to buying standardized products, and the factors other than IBM unbundling that led to the creation of the software products industry.

[Interviewer's Note: This interview was recorded during an ADAPSO conference held at the Houston Galleria, Houston, Texas, in April 1986.]

Founding of ICP (International Computer Programs, Inc.)

Luanne Johnson: So tell me for the tape how you got the idea of publishing a catalog of software products. Tell me where the idea came from.

Larry Welke: You want an honest story, or...?

Johnson: Well, you know, make it colorful.

Welke: I'd gone with Merchants National Bank at the beginning of 1965 and was put in charge of their customer services and promptly began selling services that we did not have, you know. And I'd go back to the programming staff and say, "I need this by the end of the month." And they'd say, "Two years." And nothing we were selling was new or different or unusual. I knew

that other banks were doing the same thing. So I figured, “Why the hell is this a problem? It's the same goddamn thing.”

Over the course of about six or eight months I compiled a list of what was available. That June of 1966 the ABA [American Bankers Association] Automation Conference had a swap room. If you had something to sell, you put it on one wall. And if you had something you wanted to buy, you put it on another wall with your requirements. It was the most popular room of the conference. I mean, there were more bankers there than in the bar. And that makes it a success, right?

Johnson: By definition.

Welke: The ABA had said that they were going to come out with a catalog as a consequence of the popularity of the room. And we figured anything that was a volunteer effort within an association, had no chance of coming out. So, we'll do it on a private basis. Two of the fellows at the bank and I formed a general partnership and did it. We each kicked in $250. And after 30 days, literally 30 days, the other two figured it wasn't going anyplace and they had jeopardized their $250 and they wanted out.

Johnson: That's a great story.

Welke: So they backed off and I never did give them their money back. I said, “I'm going to keep that. I just won't obligate you for anything else.” So that was agreed and that was it. I started it and ran it part-time for two years. But then by the summer of 1968 it became very evident that IBM was going to unbundle and that it was going to have a dramatic effect upon buying and selling software.

Not that it had not been tried during this two-year period. There was a lot of entrepreneurial activity going on as people were trying to find out what you could buy or sell. Or what did you have to do if you were going to resell it and use it in a second environment?

I think sometimes we forget about the fact that here it is late 1960s and we still had not gotten to the concept of reusable software or something you could use in a second environment. Everything we were doing was so tailored to the hardware that we had at hand that you could not literally use it in a second environment without modifying it somehow. What a change. That's only about 20 years ago.

Early Software Products

Johnson: I'm finding that fascinating. One of the things that's coming out of talking to all these people is that I'm trying to home in on the creativity involved in coming up with the concept of what a software product was in the first place.

One example is the way that John Postley [of Informatics] came up with the concept to think of Mark IV as a product per se which at the time was very unusual. Compared to Walt Bauer who was saying that one of the things that Informatics was going to do was develop proprietary products but he was thinking of it primarily as a tool to be used by professional services to make it more effective and efficient, to provide custom programs. Then Postley came in in 1965, 1966 and he had a very clear idea of what a product was.

In contrast to that, Dick Thatcher's Atlantic Software was formed January 1, 1968 with the idea of taking code that somebody had written and telling them, "Okay, you can recoup your investment. We're going to broker it. We're going to sell it to somebody else." It didn't work because the code had not been written with that concept in the first place. The effort to productize it and make it applicable to a second and third and fourth installation was too great.

And going back to John Postley again, his concept of what a product was was quite broad. It's not just source code. It was the documentation. It was the training. It was upgrades for *free*. For years, they provided upgrades for free.

So one of the things I'm trying to zero in on is the concept of what a software product is as opposed to a computer program. And how that evolved out of what people were doing.

Welke: There are a couple of people that you ought to talk to in addition to Postley. One would be Philip Hankins. I think Philip is with Wang at this point. I am not sure. Because he sold his company initially to Wang and then Wang sold it off. Or maybe Hankins is still someplace in the Cambridge area in Boston. But Philip was the one that took a product, a payroll system...

Johnson: Yes, I remember the Hankins payroll system. PHI Payroll.

Welke: Yes. It was from a bank in Buffalo, Marine Midland. He took their payroll system and resold it. It was a payroll system that the banks were using as a servicing package. So there was a degree of standardization, anyhow. Plus there are legal regulations governing payroll which force standardization. I don't think at the beginning anybody really thought through what was a product and what was not a product or why it could not be a product or why it is more likely a product. It was just sort of happenstance.

Phil happened to have a good one with payroll. But I don't think anyone sat back and said, “Gee, this is something you can do with payroll but you can't do with general ledger.” Another guy you ought to talk to is Bob Head. Bob is currently with the Department of Agriculture in Washington.

Johnson: Thatcher mentioned him. He was the guy that started the company in L.A. that preceded Atlantic Software by a few months. Is that right?

Welke: Right.

Johnson: It started out as a brokerage firm. They weren't developing products, they were just going to broker software that others had developed.

Welke: Exactly. They were going to take anybody's program and sell it to somebody else. And he went bankrupt so quick it wasn't funny. And there was one other guy, Dick Jones. Dick is currently in Los Angeles with Ron Davis and a micro operation. Jones was President of ADR at the time that Autoflow became a product. Now, Autoflow's a little bit of a different animal in as much as it was a way to finish up a government contract. I think it was a Navy contract that they were a little bit behind schedule on. And part of the contract called for documentation...

Johnson: Including flow charts.

Welke: Yes. And they didn't have any.

Johnson: They hadn't done them ahead of time so they were going to do them afterwards.

Welke: So some smart programmer said, “Listen, we'll do this automatically”. He wrote the program and generated the documentation. And the Navy said, “Well, it's good enough for government work. We'll accept it.” And the program that did that was then put back on the shelf. Six months later the ADR gets itself into a bind on another contract. Somebody remembers that it's up on the shelf and decides to do the same trick all over again. Then they began using it on everything. And then finally somebody in the government said, “We want to buy that.” That had never occurred to anybody in ADR. And it required a whole new decision-making process. I've got to believe that Dick would remember that. Because he used to tell this story at the ICP seminars on how to buy software. He'd tell that story on how it started.

Johnson: Let me ask you another question to get back to my first question about...

Welke: I'm sorry.

Johnson: That's all right. This is exactly the kind of digression that I want because this leads me to other sources and that's what I want. But I did have one question that I very specifically wanted to ask you about the forming of ICP. Why the hell did you call it International Computer Programs?

Welke: Because we wanted to think big? I don't know. I really don't know.

Johnson: You don't remember.

Welke: No.

Johnson: You just had the feeling that the potential was there?

Welke: Yes. Well, see when we started, we were going to compile the list just for banking. And then as we got into it we kept saying, "Well, if it works for banking it works for anything. It will work for manufacturing and distribution. And if it works in the United States, it will work in Europe." We really had most of the profits spent well before we ever took out the charter for the corporation. We had a grand plan.

Johnson: How did you initially get the word out to the vendors? That very first issue of the ICP Directory had a lot of banks listing their software for sale. I can see where your contact was through the ABA on that. There were also some service bureaus listing products in there. And then, of course, the two universities, Michigan and North Carolina. Then with each issue as I went through them I saw how there were more and more independent software vendors. It was a few years before we really began to see companies that looked like software product companies. But how did you go out and find the products to list? I mean, the thing was useless as a subscriber-based service unless you got a lot of things listed in there. How did you find all those?

Welke: Direct mail to any kind of a list we could get our hands on. Plus watching and reading all of the trade press that might have been in existence and just picking names and writing personal letters. I used to come home from the bank at five, five-thirty, eat quickly and then work until midnight. It was great fun.

Johnson: So you really just sat down and just researched every source you could think of to find who was out there that might have what looked a program product in order to get it listed. A lot of nitty gritty work.

Welke: Exactly. And a lot of people saying that it was the dumbest thing in the world. It would never work.

Johnson: Obviously there was something. There was a need for it clearly.

Welke: Yes, it wasn't smarts on my part. It was just evident. I mean, it had to happen.

Origin of the Million Dollar Awards Program

Johnson: Let's skip ahead to the Million Dollar Awards. The first was in 1971. Tell me what the thought process was behind that and how you went about it.

Welke: Another one of those very direct stories. There was a consultant named Stone, Jim Stone. He worked for Quantum Sciences, a market research firm at the time. I knew Stone. I knew Quantum Sciences. I was in New York and late one afternoon I called on him to see if he wanted to get a drink. We went to some saloon and he said, “Are you still fooling around with that software crap?” And I said, “Wait a minute, wait a minute.” He said, “That's not an industry. That's not going anyplace.” I asked him to defend himself on that. And he said, “Name something, name anything that has sold a million dollars. You got a million dollars in sales, then you've got a product. There's nothing that's done that”. I said, “Bullshit. I know that there are.” So I went back to the office and counted up 29 products that had.

Johnson: How did you know that?

Welke: By checking into the directory and by calling around.

Johnson: So you looked for products that you knew were selling a lot.

Welke: Yes.

Johnson: And you called the companies up and asked them.

Welke: Yes.

Johnson: And you said, “Have you sold a million dollars worth of this?”

Welke: Yes.

Johnson: And they always said yes, right?

Welke: Well, we asked for substantiation. We didn't to begin with, I have to admit. I think we even sent out a letter to probably 50 or 60 software firms that we thought probably had done it. I mean, if it's somebody who just started, you know that they wouldn't have done it.

Johnson: Unless it was John McGuire [of Software AG].

Welke: Yes.

Johnson: He only had to sell four [at the prices he was charging].

Welke: Once we had the list compiled then we sort of concluded, “Well, if we didn't know this, nobody knows it either. And we're trying to pump software product and the marketplace.” I mean, it was very self serving. But what the hell? Why not do it? Let's go tell the business press. Let's do something about it. Give them a non-edible plaque. So that's what we did.

Johnson: And how many awards were there in 1971?

Welke: Twenty-nine awards.

Johnson: The first time you gave 29 awards.

Welke: But we didn't think anyone would ever come to get an award, so we wanted to make a show out of it. We handed them out at an ADAPSO meeting down in Miami.

Johnson: The first one that you did on your own, wasn't that in New York?

Welke: No.

Johnson: Chicago?

Welke: It was both. We had L.A. and San Francisco as well.

Johnson: Because you didn't think they'd come to you?

Welke: Nobody would. What the hell -- make it easier, you know? So we purposely staged these meetings all the hell over the place. We did that for two, three years in a row.

Johnson: I can remember when you had east coast, west coast events. I can remember going to those. But I didn't realize initially that you had it that spread out. When did this become an attraction to the point where you felt that you could get people to come to you?

Welke: I think it was 1975. We finally held one in Chicago, a central one in Chicago.

Johnson: I remember being at that one. I remember Tom Nies [of Cincom] standing up at that meeting and saying, “We have met the enemy and he is us.”

Welke: Good guy.

Origins of Advertising-Based Publications

Johnson: When did you first start the Skinny, the advertising-based publication?

Welke: 1972.

Johnson: Clearly at that point you felt that the industry had developed to the point where there was a market out there so you could get people to pay for listing these things.

Welke: Right. God, I'm telling you stories out of school.

Johnson: You and a lot of other people.

Welke: We started holding seminars on buying and selling software in 1969. In 1970 there was a credit crunch that just closed down the market. And where we had 200 – 300 people

attending seminars in 1969, we had 12 – 13 attending in 1970. We had overcommitted, prepaid the hotels, and the speakers and we were on the verge of going belly-up. So we had a creditors meeting when my indebtedness exceeded my assets by $480,000. I mean, we were dead in the water. So I told the creditors, if you want to wait around you'll get all your money back but it's going to take me four years to pay it off. If you want it right now, you're going to get four cents on the dollar. So they had no alternative and they lived with it.

Then we went back and got rid of half the people in the company and sat back and said, “Now what are we going to do? How are we going to make this thing work?” And after a little bit we determined that what we should do is come out with a brochure. But we didn't have any money to do that. So I sat down with a friend of mine, Sam. I forget his last name. Sam and I were having a drink up in Chicago. I said, “The problem is that our product is so complex you can't describe it easily. I really need a full-page ad or a full-page brochure to do that.” He said, “Fine. I know how you pay for it. You're talking about software products. What you do is you talk about your directory on one side and you say, ‘For an example of what is in the directory, turn this over.’ And you give a sample listing. Only you don't give that sample listing away. Because you're promoting an actual software product by doing that. So you tell the vendor that they're going to have to pay to get on the backside of your brochure. You just charge them enough to cover both sides of the goddamn brochure. That guy pays for all of your promotion.” Nifty idea.

So I went back to the marketplace and figured if it works for one program maybe I can get two programs and list them, or maybe three, and really make a little bit of money on this thing. By the time we had called everyone, it was a great idea. It just took off. And before we knew it we had 32 pages in a brochure. And 32 is sort of a magic number in printing because it has to be multiples of 4, 8 and 16. And we figured, okay, we can take 4 pages of that 32 and describe anything we want as far as our product. But, unwittingly, we had devised the first Skinny. I mean, bang. There it was. And it just took off. And we figured if it works once it will again. And it did.

The problem was at this particular point no one knew how to advertise software. I mean, how do you take a picture of a payroll system? But a program listing, that made sense. Everybody could understand that. It was very easy to describe in a formatted form what your product was. I think that is what caused it to be such a success.

Johnson: That particular format that you came up with through the directory turned out to be one of the best advertising mechanisms that the software vendors could use.

Welke: Exactly.

Software Product Pricing

Johnson: One of the things that's fascinated me is the process that people went through trying to figure out what the heck a product was. I mean, they all thought they knew. Nobody sat down and said, “Well, what is a software product and can I make one?” They sort of said, “Gee, what I've got *is* one.” But then everybody went through the process of figuring out how to price it, through some very different channels of thought. Give me some of the things that you remember about things that people went through and the logic behind it. Everybody talks about John McGuire coming in with this real high-end price in the early 1970s and saying that this is the way that things should be priced.

Welke: Quarter of a million dollars. All the way down to... who's the guy that was running Boole and Babbage at the time? Ken Kolence. And Ken stood up in front of the group one time and went through an elaborate equation on how he set prices. You have to know Ken. He's an engineer. He's incredible. I think he takes a whiz break according to a formula. He laid out how you do the costing and you do the market estimation. You divide by the potential number of units and everything.

Johnson: Very traditional manufacturing approach to it.

Welke: Yes. And somebody in the audience said, “What’s the most recent product that you've brought out?” And Ken gave them the name of a new product. And the guy said, “How did you set the price?” He said, “We pulled the number out of the air.” Wait a minute. Blew his entire presentation.

I think the most common formula was you charge one-tenth of your development costs. But as I also recall very few people had a standard on what constituted development costs. What is that? Is that designer time? Is that coding time? When does coding stop and maintenance begin? Those questions were never answered. It was just one-tenth development cost.

Johnson: One-tenth of what it costs you up to the point when you started selling it.

Welke: Exactly. And that's what it netted out to. Which is obviously just bullshit.

Johnson: Informatics has done a history of Informatics. It's a fairly thick document. Frank Wagner hired an historian to research and write it and included the story as to how they came up with the price of Mark IV. What they had done was hire at least two different consultants. They hired Brandon Applied Systems who did market research for them. And John Postley told

them what he thought it should be. They consulted several people and they had a price range that went from something like about $15,000 up to about $45,000. So they split the difference. It came out at $30,000. I mean, it was really just a very seat-of-the-pants kind of process that occurred.

Welke: It was always the same thing. For instance, remember back when there was a two-tier pricing scheme between DOS and OS products?

Johnson: Yes.

Welke: There wasn't that much difference between the goddamn code and no relationship to functionality. Why should OS cost $10,000 more? Literally 30 percent more. Because, what the hell? It was a bigger machine.

Johnson: Bigger machine. It cost them more to buy the machine. So therefore, it should cost them more to buy the software.

Welke: Yes. Or it was a larger customer and they should have more money.

Johnson: Yes.

Welke: Now we got to the real formula called value pricing. What the market will bear. There was also some competitive pricing. At our seminars, we always pointed out you either do a good cost study and you know what your cost of development is and what you have to recoup and then what your marketing costs are and do your estimated market and all that bullshit. Or you do a competitive study and find out what your competitors are doing. Or you do value pricing and peg it whatever you want. But competitive never really worked. Because you can for the most part... Not anymore but back in those days, you could define the product however the devil you wanted to define it. Accounts receivable was not accounts receivable. Yours was whatever it was and mine was...

Johnson: Whatever.

Welke: Yes, yours had balance forward and open item. And mine just had balance forward. There's so much variation that the seller defined it. However, there were cases where the buyer defined it with an application such as payroll. With payroll, the buyer knew what he needed. And either you had it or you didn't have it. You couldn't go in and say, “Gee, you can get by with

this." Because he'd come back and say, "No, I can't. My payroll must do this." And if you didn't do that...

Johnson: Well, it wasn't the buyer so much as the external forces that were in place.

Welke: Exactly.

Johnson: The regulations and union requirements.

Welke: Yes. So then you got Philip Hankins payroll and the MSA payroll and the Computer Sciences payroll. Then there was another one out of Huntsville, General Computer Services.

Johnson: Yes, GCS.

Welke: And all of those payrolls were competitive. And they were all within a couple thousand dollars. And they would watch each other as far as who was charging more. They would start out. Everybody was sort of at $16,000, $17,000 or $18,000. Someone who was bold enough would go up to $20,000. And everybody would climb up right behind him staying five dollars...

Johnson: At least as far as the list price.

Welke: Yes.

Johnson: But with those particular vendors... I'm not sure about GCS but certainly with MSA that price was always negotiable in the actual sales close situation. Which brings me to the point that looking through those ICP directories in the late 1960s, early 1970s, there were an awful lot of people that did not list their prices. And I assumed that they wanted to retain the option of changing it at the last minute, that they thought they were going to lose the sale otherwise. Thatcher claimed that was not the case in their case. He said that they had set prices and they did not negotiate. They just thought somehow if they didn't publish the price that their competitors wouldn't know what it was. He said it was actually very childish. Because all your competitor has to do is find out from the customer you're trying to sell to how much you're charging.

Do you have a feeling for how many of those people that didn't list their prices it was because the prices were soft? I mean, it was really very evident in those days that a lot of people were not going to put their price in the directory listing.

Welke: There was a lot of misunderstanding as far as where or how you made your money in the software product business. And it was not clearly understood by even some of the companies who were pretty substantial players in the marketplace. There was that comprehension that, “Gee, the cost of the product really is the cost of reproduction once the product is established. So just once we can discount heavily.”

Johnson: Yes. “It's only going to cost us five dollars to make another tape.”

Welke: Because it would be nice to get the user's name on the list, right?

Johnson: Yes.

Recognition of the Value of Maintenance Revenues

Welke: You do that, that's just death. I don't think it was until 1972, 1973 before people began to recognize the value of a maintenance revenue stream on their income. And we look back at it now and say, “God, that's so evident.” Wrong. It was not evident, not back then.

Johnson: Informatics didn't see it. Informatics provided upgrades and enhancements free for about five, six years after they first started shipping Mark IV. It wasn't at all evident to them that was.

Welke: Exactly. I can remember the discussions we had with MSA advising them on that entire thing as far as what the industry was doing. I remember [Bill] Graves was saying, “You can't possibly charge 3%. You cannot. I mean, nobody's going to pay that.” But they did. Everybody paid that. Nothing to it. I think there was a lot of price, as you so nicely put it, easing. What was your expression?

Johnson: Soft pricing.

Welke: Soft pricing.

Johnson: But I think your point is good that they didn't really understand what that price was based on. There was no clear understanding of how they arrived at the price in the first place so, therefore, for the salesman it was real easy to slip around that.

Welke: Exactly right.

Impact of IBM's Unbundling

Johnson: I got started in this whole project because of working on an article about IBM's unbundling. And you said a little bit ago that in 1968 it was very apparent that IBM was going to price separately and that there was going to be a big demand. Tell me what you remember. One of the things that became clear to me when I first started looking at the unbundling and the impact of that was that the unbundling was something that happened within the context of a lot of other things that were going on at the time. And without understanding all the other things that were going on it was going to be hard to determine what the true impact was of unbundling, to what extent that truly created an industry. Clearly there was this perception out there before the unbundling that...

Welke: Well before.

Johnson: That it was going to have a tremendous impact. The myth still persists. Here is the first and probably the last issue of *ADAPSO Research*. Embedded in this is a statement that says the surging industry revenue in 1969-70 was in large part due to IBM's unbundling of software. And then you look at the chart. Look at this surge here.

Welke: Oh, watch that surge. Nothing, right? Boy, it's surging right along. Wow. That's funny.

Johnson: So this myth persists. It's going to be fun interviewing Lee [Keet] because he put it in his book. About page 15 of his book, he says IBM created the software industry. This myth persists. So go back to your memory of your perceptions at the time as to why you thought it was going to be worthwhile to start ICP because of this great impact. And why you now say, "No, it's not true, unbundling is not what created the industry." It obviously played a role. But I'm trying to fit that role within a broader context.

Welke: Well, Dick Jones at the time was selling a sort system against IBM's sort. But IBM's was not priced. And he wanted IBM to separately price. But you could easily also find other people, particularly in the applications area, who never raised the question of IBM pricing as being an issue that anybody wanted.

Johnson: So the perception was very different for the systems software people than for the applications software people.

Welke: Yes. But I think that the marketplace would have developed whether or not IBM had priced separately or unbundled. I didn't see the market take off as a consequence of IBM

unbundling. There was no dramatic increase in sales or company formation or anything like that.

Johnson: It's not your memory that suddenly in 1969-70 there were dozens new software companies that sprung up.

Welke: Not at all. I think the street was aware of IBM's coming unbundling nearly a year before it happened. It was common knowledge.

Johnson: Everybody knew it was going to happen. As a matter of fact, I researched the business and trade press during that period of time, the articles in *Datamation* and so on, and what was really fascinating was the fact that everybody knew it was coming and everybody was busy predicting how much they were going to cut their hardware prices.

Welke: Yes.

Johnson: And the predictions were everywhere from 15% to 30%. And when they announced it was three percent, then it was fun. I have a nice collection of articles appearing just after that screaming and yelling.

Welke: Talking about the price increase that IBM just imposed on the public.

Competing Against Software Developed In-house

Johnson: One of the things that I've been hearing people say is that the applications people did not really see IBM as competition. But there's a lot of talk that the problem that people faced initially going out there was selling against the customer's belief that he could do it for himself much better than you could do it for him, that his needs were so unique that he wasn't going to buy a software product. He was going to do it himself, write it himself. So that perhaps there was an impact that IBM had in terms of legitimizing the concept of software as a product by virtue of this separate pricing.

Welke: That's true.

Johnson: And you think that that flows over into applications? Or is that really again still the system software? Is the IBM unbundling impact really on the system software side?

Welke: I would think much more so on the systems side. There are still 50% more or less of the users out there that don't think they should ever buy application software.

Johnson: But they will buy systems software.

Welke: Systems software? Sure.

Johnson: They're not going to write their own systems programs.

Welke: It's cheaper than getting systems programmers and we don't know where the devil you get them anyhow and they're very expensive.

Johnson: You say 50% would not buy application software products.

Welke: Yes, easily. This is on the basis of research we did just last fall.

Johnson: Are you including micro or not?

Welke: No. It's not micro. Micro's a different game. I've got to believe that that is nearly 100% software products versus doing it yourself.

Johnson: What you're saying then is that your research is really oriented towards the corporate data center.

Welke: Yes, exactly.

Johnson: And, of course, the people that are buying the micro software are not the corporate data center. That's the guy down the hall who's tired of waiting for the corporate data center to do something.

Welke: Exactly.

Johnson: So you found that it was 50% of users, and this is as of the fall of 1985.

Welke: You betcha.

Johnson: I can remember very clearly the point early in 1974 when I suddenly realized that I was selling against other software vendors as opposed to against the customer's idea that they could do it better themselves. And I rejoiced.

Welke: Yes. Major dramatic change.

Johnson: Yes, because now it was a matter of selling feature by feature, price per feature against another vendor as opposed to going in and trying to convince them they were foolish to try to do it themselves.

Welke: That's right. I mean, now you had something to hit against, right?

Johnson: Because when they're writing it themselves there's no way that you can possibly compete on the feature-by-feature basis because they're going to write the features they want. So then you try to convince them you're competing on a cost basis and their costs are real soft and they don't know what it costs them to do it half the time.

Emergence of Software Developed Specifically as a Product

Here's something that was interesting in the February 1973 issue of the ICP newsletter. You stated that 49% of available product was developed as product. And as recently as four years ago it would not even have been 9%. I love these figures. I'm sure that you have substantial documentation of these figures.

Do you have any further comments about that? Between 1969 and 1973, you apparently saw a great deal of development of product as product as opposed to being developed for a specific user.

Welke: People very quickly learned you can't play that game of take somebody else's program and make it a product. I used the example, "You can take a hotel room and make it into your living room. You can't take your living room and make it into a hotel room. It just doesn't work." There were so many horror stories of people who tried to take somebody's program and turn it into a product. The goddamn architecture was wrong. The structure was wrong. You just couldn't do it. This was even before structured programming was the thing in the user's market. In actuality, structured programming was used by the software vendors well before it became popular in the user end of the market. You were doing structured programs.

Johnson: Sure.

Welke: You didn't call it that because there was no goddamn name for it. But you couldn't develop a product if you didn't have a structured approach to the damn thing.

Johnson: There's another aspect of that. Because we all know it costs a lot more to develop that kind of product than it does to write a program.

Welke: Yes.

Johnson: So clearly these companies that were developing this 49% of the products that had been developed as products had an awareness that there was sufficient market and opportunity to justify that investment. So what changed that perception in the four years? Are we back to IBM again changing that?

Welke: I don't know. Good question.

Johnson: We may not ever know. It may just be the fact that the time had come. When the first computer was built, nobody thought in terms of how many software products to build around it. That just may be directly tied to the time lag between the time the 360 was first delivered. That may be what that's all about.

Welke: That's an interesting thought.

Johnson: Because clearly by 1970-1971, people were looking to make money selling software products.

Welke: Yes.

Johnson: It wasn't just a matter of: “Hey, we've got this thing we've already developed. Let's see if we can't recoup some of our investment.” It was: “Let's put some money into developing something because we're going to get our money back.” And I'm trying to figure out where and how that was changing. A realization was developing that there was an industry there and that people were going to make money at it. And people were going to try to get on the bandwagon.

Welke: That indeed it was an industry, yes. I don't know. It may presumptuous, but I've got to believe that a little bit of it might have been because of things such the publicity of the million dollar products in 1971, that there actually are such things. Depending on what the hell it is,

you can sell a substantial amount of something. And we did begin getting a couple of success stories. Mark IV was a success story in 1971, 1972. Pansophic's Panvalet was a success story.

Johnson: What do you know about the starting of Pansophic? I'm trying to figure out if I can find the first company that was formed specifically to develop and market a software product.

I've pinned this down a little bit because we're beginning to find people who were out there marketing products. Like Atlantic Software was formed both for professional services and to broker product. But that's not the same thing as committing the development expense. The same thing with Software Resources. They thought they could market software products.

Welke: Yes.

Johnson: But I'm trying to figure out where we start coming into companies that are...

Welke: Formed specifically to develop and market a product.

Johnson: Yes. I don't know if it's really that significant. I mean, maybe all the companies evolved out of something. I don't know if it's really that important.

Welke: MSA certainly was started as a professional services consulting firm. But when John Imlay took it over and cut staff that was a clear declaration for product. Pansophic you said was started really as professional services?

Johnson: No, I don't know about Pansophic. I was just asking you if you know. I've got Joe Piscopo on my list. I'll be talking to him.

Welke: He had been a programmer for, I think, Sears & Roebuck or something like that and was convinced that he could do a library system. Cincom was out there very early too. They started as a contract programming firm.

Johnson: Yes, Cincinnati Computer. I just found out that's where Cincom came from. As I get into this more and more and think about it more and more, what's most significant is the point at which people really begin to feel that they could invest the money to develop a product. That's a lot bigger decision than saying I've got a product and I've used it a couple of times. I

can maybe put some documentation stuff around it and begin to sell it. It's a different thing to say that I'm going to make the investment to develop this product because there's enough market out there that I could sell it.

Software Products as a Business Like Any Other

I've got one further thing for you. And then, of course, if you want to digress on any other thing I'm always happy to listen to it. You made a comment when we met in Indianapolis about people getting over the idea that this industry was different. Your statement was this industry's no different than any other industry. Could you expand on that a little bit? And why you even felt it was necessary to make that statement, why you think that mystique has arisen that this industry is so unique and has special problems.

Welke: Well, because when you look at the industry and what it has gone through you find that all of Business Administration 101 applies to this industry. All of the things that we learned in product management, etc., are as equally applicable to software as they are to Proctor & Gamble or anybody else that's in the product business. There is nothing unusual about this particular business.

Johnson: You don't think that there is something unique about a software product? Going back to this old debate about whether it's tangible or intangible, whether it's a product or a service, you don't think there's something about that product that creates unique problems for this industry?

Welke: I don't really think so. We talk about the special relationship that you have with your client because of the requirement for maintenance, because it's a living product. You've got that with any good product. And if you're managing your product and your marketplace and responsive to your customers, you're doing it when you're producing automobiles or anything complex. Not wheelbarrows or stepladders, you know, because what can go broke on one of those? But if it's a complex product you've got to have that kind of relationship with your customers. Anybody who produces a living product does. And I would submit that there are a lot of living products. Software isn't unique in that respect.

Johnson: Do you think that just high technology products are limited to that? What we call high tech, computer-oriented, communications-oriented products.

Welke: No, no. I think the automobile is a perfect example. Not that that is not a high tech product.

Johnson: Well, we use the term high tech in a very specific way in this industry. One of the things that people talk about a lot is not directly related to software products. People talked about the fact that there was a lot of public money available for software firms, which of course were professional services firms, in the mid to late 1960s. And then that all went bust. And there was a big scandal about the amount that had been capitalized because those firms were putting a lot into R&D in software product although they had not been formed for that business. Which sort of implies that there was a lack of business acumen among those people. This kind of comes back to your statement. We learned all this in Business 101. Maybe the problem was there were an awful lot of people who hadn't had Business 101.

Welke: How many firms do you know that were started by technicians, by good programmers, by people who didn't have the foggiest notion of what business was all about? I can think of hundreds of examples literally where the technician was the driving force behind it and didn't know beans about finance or marketing or management or anything like that.

Johnson: But in that sense it probably is no different than any other start-up industry. In the same sense that when there were several hundred automobile companies, the guys that were running them were engineers.

Welke: That's right. But I think you still had the same phenomenon of a lot of investors that were hurt because they were putting money into things that were not being well managed. Not that the technology was bad. And there were a lot of horror stories.

Johnson: Well, that’s what makes this project so interesting and so much fun. Thanks so much for taking the time to give me your perspective.

Welke: Thank you. And good luck with the project.