RAMIS and NOMAD--National CSS

Moderated by:
Burton Grad

Recorded: May 20, 2005
Westport, CT

CHM Reference number: X3738.2007

Table of Contents

Conference Call: RAMIS and NOMAD--National CSS

Conducted by Software Industry Special Industry Group – Oral History Project

Abstract: Five of the National CSS principals participated in a recorded telephone conference call with a moderator addressing the history of the company's use of RAMIS and development of NOMAD. The licensing of RAMIS from Mathematica and the reasons for building their own product are discussed as well as the marketing of RAMIS for developing applications and then the ongoing revenue from using these applications. The development of NOMAD is discussed in detail along with its initial introduction into the marketplace as a new offering not as a migration from RAMIS. The later history of NOMAD is reviewed, including the failure to build a successor product and the inability to construct a viable PC version of NOMAD.

Participants:

Name	**Affiliation**
Dick Bayles	National CSS
Judd Boykin	National CSS
Harold Feinleib	National CSS
Nick Pisarro	National CSS
Nick Rawlings	National CSS
Burton Grad	Moderator, CHM
Karol Hines	CHM

Burton Grad: This is an experiment to see whether we can get quality information using a conference call and transcribing the recording. The transcription will then be posted on the IT Corporate Histories Project website.

Nick Rawlings: Will you give us an opportunity to retract or edit what you put up on the web?

Grad: The answer is yes, within reason. We're trying to see if we can get quality historic information from five of the people who were most closely involved in RAMIS and NOMAD, particularly over the first ten years of National CSS, the 1970 to 1980 period. I will be the moderator and may ask specific people to talk, because if multiple people talk at the same time, we don't get a clear recording and we can't transcribe it. We will also ask you, as you speak, to give your name first, and with the two Nicks on here, it'll have to be a Nick R., and a Nick P., or however you choose to do it. That way the transcriber knows who's talking. The plan is to spend the first 30 minutes or so talking about RAMIS; we're not going to just go through the checklist of questions [which are included at the end], but we're going to talk about it in a conversational manner. At the end of the 30 minutes, we'll check quickly to see what items we haven't covered and go back over them. And then we'll switch over to talk about NOMAD, with another 30 to 40 minutes on that subject. In the last 15 minutes or so, we'll try to open up any subject you think should be discussed related to topics that we haven't covered, and to ask you what other products were produced by or supported by NCSS?

Any questions? Please identify yourself, and don't step on the other guy who is talking. Is there anyone who has a specific question they want to ask before we start? Okay, you know it's all being recorded. We will then post it. But you will have a right to edit the transcript before it is made available to anybody else other than those of us involved here, okay?

Let me ask a question first of all: who would be the person to start with on RAMIS. Who made the decision that NCSS should go ahead and get RAMIS as a product?

<u>Licensing RAMIS</u>

Harold Feinleib: I'm not 100 percent clear, but I think it was Dick Glazer.

Dick Bayles: I would agree. I was going to say the same thing.

Grad: Harold, I know you tried to get Dick Glazer on the call. He said he couldn't make it today. But I'll follow up with him later. So you think that he was the one. Do any of you know how he got started, or why he got started with RAMIS?

Nick Pisarro: Dick Glazer was in charge of bringing third party products into NCSS and getting them marketed. And so he was in charge of doing that. He sort of just went out and tried to get anybody and anything that would agree to our marketing their products. So it may not have been that deliberate a plan to include RAMIS, though once it was onboard, it was very popular.

Bayles: If we talk about other products that we offered, some people may be able to think of a lot more, but the only other one that comes to mind in the same genre as RAMIS was InfoTab which was a spreadsheet system that came out at about the same time as RAMIS did.

It was owned by a different organization. But again, it was Dick Glazer; I guess his title was Product Marketing, probably, something like that. And getting third party people to come in was what he did.

Grad: Do any of you know anything about the negotiations for RAMIS, the setting of the initial royalty rates, or any of those kinds of things?

Everyone: No.

Grad: Okay. We'll have to pursue that with Dick Glazer to get the beginning. Harold, what was your first connection with RAMIS and when did it occur?

Feinleib: When we got connected with RAMIS, we started selling it on our time-sharing system, and people were using it and liking it. And that was what gave us impetus to build NOMAD later. Also Mathematica was becoming difficult to negotiate with. I think we had some difficulty with them. We felt that they were more interested in selling the product as a product, rather than what we were doing on the time-sharing server.

Judd Boykin: I recall that Dick Orenstein, I believe, was negotiating with Mathematica when our contract was running out, and they wanted a much higher percentage of the money that RAMIS generated than we had been giving them. And Orenstein got very, very upset about it, because he felt that they were taking advantage of us, because we had so many customers using it. And he finally agreed to a higher rate, but that's when he decided, "We've got to replace RAMIS, and we're going to build our own product."

Grad: I'm going to ask you all to back up now. You're talking about the decision to make NOMAD. That was two or three years after you started offering RAMIS, wasn't it?

Boykin: The original contract, as I remember it, was for two years and then we renegotiated for two additional years.

Grad: So let's go back to 1970. Dick Bayles, was there any technical involvement in RAMIS by NCSS?

Bayles: That's an interesting question. Does anybody remember what they programmed that in?

Rawlings: Yes, I do. They programmed it in FORTRAN, and indeed, they were one of the early users of our FORTRAN debugging system, and I remember lots of comments from them about how great it was, and how it made it a whole lot easier for them to add facilities. I know that we did work to make RAMIS run faster on our system, and also to make it more secure, so that people couldn't steal it, or bypass the surcharging mechanism. So we made changes to our operating system. I've forgotten now what they were, but I remember working on them. Also, I seem to remember that Harold worked with Gerry Cohen on some stuff associated with file system performance.

Feinleib: I'm impressed by everybody's memory.

Bayles: So am I!

Grad: Let's go back to 1970. I understood in some of the conversations that there was quite a bit of technical work that you all did to make improvements to RAMIS

Bayles: No, only to the underlying systems that supported it. I suspect we never even saw any RAMIS code.

Rawlings: I believe that's true.

Feinleib: I agree.

Grad: So in fact, you're telling me that as far as RAMIS is concerned, you basically stayed on the outside of the system. You didn't actually look at the RAMIS code itself?

Bayles: That's correct. And that was true of virtually every third party product. One of our theoretical attractions, and in fact, practical attractions, was that we could offer close to a true IBM operating system and language environment. So that if somebody had developed an application for the 360, then with relatively few modifications, if any, depending on how complicated it was, they could run the thing on our system without us needing to be involved, unless there were special charging issues, or something else. But I think we virtually never got involved in the third party's code.

Grad: So you truly treated this as Mathematica's product that you were offering on your time-sharing system.

Feinleib: Absolutely.

Grad: Okay, a question. As I say, I had the impression from someone, maybe it was from Nick Rawlings, but maybe I'm wrong on this, that when you, later on, went to do NOMAD, one of your concerns was that you had made some improvements to RAMIS, and your question was whether they belonged to you, or whether they belonged to Mathematica.

Pisarro: I think what was said was that we worked with them to make some changes in the underlying operating system, CSS, whatever it was called, in order to make RAMIS run more efficiently.

Rawlings: And also to make it more secure so that people couldn't steal it. Because we did recognize that that was an issue. So we made changes to provide increased security, and to improve performance. But they were to the operating system.

Grad: So, it was not changes to RAMIS, it was changes to VP/CSS?

Rawlings: Yes.

Grad: To make RAMIS be more secure and more effective.

Pisarro: We would do that to help support these products better. We would make changes to the operating system, but we didn't change the products themselves. I think there were some technical issues in terms of Mathematica's responsiveness to customer needs, that we felt weren't being met by Mathematica, as well as licensing issues.

Bayles: Correct. I believe that was true.

Rawlings: Let me jump in. We felt that their reference material was terrible, so we rewrote it. We rewrote the classes. We taught the classes. And in teaching the classes, and in providing professional services, consulting to our prospects, our joint customers, we ran into a lot of situations where the customers said, "If we only had this feature, it'd be much better." And we did pass that information on, and were disappointed with Mathematica's lack of responsiveness.

Grad: Can somebody tell me who was the primary interface that you worked with at Mathematica on technical issues?

Rawlings: Gerry Cohen.

Grad: So this is Gerry Cohen before he started his own company?

Rawlings: Yes.

Feinleib: Maybe that's why he was not responsive.

Pisarro: When did Information Builders start? Anybody know?

Rawlings: Information Builders started about a month before we released NOMAD, I think it was in September of 1975.

Rawlings: And I think my recollection is that when they started it, they had a product that was 100 percent code compatible with RAMIS and it had the same bugs and the same misspelled error messages.

Grad: So Gerry was with Mathematica through 1974?

Rawlings: I think through 1975.

Grad: What was the primary use of RAMIS by your customers? Who would want to speak to that?

Rawlings: Basically, they would set up a single-user, large set of data. Possibly multi-level, single-path hierarchy, and produce reports of various kinds off it. Loading new data in, maybe monthly, and producing new reports. Very important to produce reports off of data, and that's what it was about.

Grad: Were they using it primarily to develop the application, or to run the application or both?

Rawlings: Most of our revenue came from their running it.

Feinleib: But they developed it, they wrote it in RAMIS on VP/CSS.

Rawlings: And often with our help.

Professional Services

Grad: Was there a professional services operation at NCSS at that time?

Rawlings: Absolutely.

Grad: Who was in charge of that?

Boykin: I think I was.

Grad: You were in charge of professional services?

Boykin: Somebody correct me if I’m wrong, but I’m sure I was.

Pisarro: I think that’s true.

Grad: So what kind of work, Judd, did you and your people do to help the customers?

Boykin: Well, they would come to us with their problem, and they’d say, “We need a database system to solve it.” And we’d tell them about RAMIS, and we would make a proposal and develop it for them. I think RAMIS development was a major part of our professional services.

Grad: Were you doing any training at that time, Judd?

Boykin: I did do training. For example, Bank of America was a big customer, and we did all their training for them. That was done in the field, though, out of the regional offices.

Rawlings: We had regional offices that were trained in RAMIS. We had a major group that was under Ruth Morley, and I think maybe Russ Gloerson and Larry Smith, also, took part in that.

Grad: And would they work on site with the customer? Could they work remotely?

Rawlings: Yes, both.

Grad: Judd, was there a significant amount of revenue coming from the professional services?

Boykin: Yes.

Bayles: But it was always relatively small.

Boykin: Yes, compared to time-sharing revenue, it was small. But I do remember the first quarter that we became profitable (in 1970), the reason that we became profitable was that we broke even with time-sharing revenue, and the profit came from professional services.

Bayles: Our objective in professional services--correct me if I’m wrong here--was to capture the ongoing running revenue.

Boykin: That is correct.

Bayles: We wanted to make money on the front end, but we were really trying to tie the customer in to the long-term execution.

Boykin: That is absolutely correct.

Grad: I have a question. Does anyone know what percentage of your time-sharing revenue was related to RAMIS, say, during the 1972-1974 time period?

Rawlings: About ten percent.

Bayles: I would've guessed maybe 15 percent, but that's in the ballpark.

Grad: So it was significant, but was not a dominant use of the machine.

Rawlings: Correct.

Using Time-Sharing for Program Development

Bayles: NCSS' reason for existing when we started was that we were a program development platform.

Boykin: Exactly.

Bayles: We wanted to develop stuff for the IBM 360 operating system, and we wanted to do it effectively and efficiently. We provided the best way to do it, and you could transport it, if you chose to, to your own machine. Luckily, not a lot of people did transport the applications.

Grad: Are you saying that during this period this 1970-1974 time period, that the bulk of your revenue was coming from the debugging on-- let's put RAMIS aside-- on other applications or the running of those applications after they were debugged?

Rawlings: It's hard to say. I think that the debugging and the running of FORTRAN applications was like 70 or 80 percent of our revenue for the first year.

Pisarro: COBOL might have been the major development tool, because there was no way to develop COBOL efficiently on any other platform.

Boykin: Except with us.

Rawlings: Right. But I think the execution of COBOL programs was not as great as the execution of FORTRAN programs.

Boykin: I agree with that.

Bayles: I also agree with that.

Boykin: A lot of the development that was done was then ported to the user's 360.

Grad: On both FORTRAN and COBOL or just on COBOL?

Boykin: I'm familiar with the COBOL side. I know some of the COBOL applications remained and were run on VP/CSS, but the majority of them just used us for the development and debugging of the system, and then would move it back to their machines for execution.

Rawlings: That was true for COBOL for sure.

Feinleib: That was because COBOL typically moved a lot of tapes around. They did a lot of tape processing, and they had the tapes in-house, and that was even probably before they did a lot disk processing.

Rawlings: Yes.

RAMIS for Solution Execution

Grad: Can I ask you how this relates to RAMIS? Did the same thing happen? Did most of the customers use you for debugging and then run RAMIS on their own machines? Or did they run the RAMIS applications on NCSS' system?

Rawlings: No, absolutely not. Almost all of it-- all of our revenue was their running it on our system. I mean, maybe ten percent was the development, and 90 percent was the execution.

Boykin: I would agree with that.

Feinleib: RAMIS was an interactive program. The COBOL programs were batch programs designed for batch processing. So once they developed it, it was much, much cheaper to run these on their own machines, and the data was local to their machines. These were batch programs, so they were better off running on the batch machine. But RAMIS was interactive, so the time-sharing terminal was the perfect vehicle for delivering RAMIS execution, because the person using it wanted to interact with his data, that's why they used RAMIS.

Grad: Explain what you mean. Go further on what you mean by interaction, because I don't think that that's what we think of today as interactive.

Feinleib: It's similar. The program puts up a prompt, you enter some data, it maybe prompts you again, gives you some output, and the user's interacting with it, and then maybe running a report. So interacting, specifying filters, that kind of thing that you would do with a typical report writer.

Grad: Was RAMIS interpretive or was it compiled?

Feinleib: Interpretive. But I also have another comment about COBOL versus RAMIS. If you wanted to run RAMIS in-house, you had to buy an expensive-- and I forget what the price was-- copy of RAMIS for in-house use. With COBOL, that wasn't the case because you just needed the compilers on your 360. In fact, you could have used the object code, but probably they recompiled on their own machine, and that was not an expensive proposition for them.

Rawlings: Initially it was free. COBOL F was free.

Feinleib: It was free then. But RAMIS, if you wanted to run it in-house, you had to pay, what was it? A couple hundred thousand dollars or something like that. A lot of money. So the people who went to the time-sharing service, VP/CSS, were people who didn't want to go out and buy RAMIS, primarily.

Bayles: You also have to recognize that in the 1969-1975 timeframe, effectively, there was no interactive IBM mainframe on an in-house basis. I mean, if you wanted to install VM 370, and maybe 30 or 40 people did, then you could get everything that we had, minus the COBOL and minus RAMIS and minus all of that. But if you were an average commercial shop that was it. TSO and the other options were god-awful.

Feinleib: You could manually write a terminal application for a specific application to run under OS 360, but it was not a general time-sharing system that you sat there and typed commands like you do, say, in UNIX today.

Bayles: If you wanted to run RAMIS interactively, you were kind of up a tree.

Feinleib: Right.

Grad: Except that they could've run RAMIS under VM on their own machine, but I gather you had an exclusive license, and were the only ones allowed to offer RAMIS on a time-sharing basis?

Bayles: On a commercial time-sharing basis.

Rawlings: They could probably have licensed it to somebody for in-house use.

Bayles: Someone, for example, that wanted to get RAMIS, they probably could've gotten a license for it. I'm not sure of that, but that's my guess.

Pisarro: But what Dick Bayles is saying is that VM did not take off until...

Rawlings: Much later.

Bayles: 1974 or 1975, in that ballpark.

Pisarro: Yes, because I was supporting VM, and it just small potatoes until the mid-1970s or later.

Grad: Was there any other competition to RAMIS? Were there any other 4GLs at that time? Did you consider Mark IV a 4GL?

Rawlings: Yes.

Pisarro: We did not consider RAMIS a 4GL. NOMAD was considered the first 4GL.

Rawlings: No, that's not true. I think RAMIS was a 4GL.

Feinleib: We weren't selling it as an application development facility that you could write an application with a language. We thought of it as a report writer and database.

Grad: So in that sense it was competing with something like Mark IV then?

Rawlings: Absolutely, yes.

Grad: Did you put Mark IV up on your machine?

Rawlings: No.

Grad: Why not? There were thousands and thousands of Mark IV users out there.

Bayles: But if they already had access to it, why would they want it through us?

Grad: So you're saying that the economic advantage was a primary factor for you in using it under the time-sharing framework. It was much cheaper for them to get it that way, than for them to rent the copy for their own use? Is that correct?

Feinleib: Yes. Yes.

Bayles: But Mark IV was available on a lot of other machines.

Rawlings: It was on GE mostly, wasn't it?

Bayles: And maybe it wasn't even 360 compatible. I'm not sure of that.

Grad: Mark IV was 360 compatible.

Bayles: It was, okay.

Grad: Yes, absolutely. They were one of my clients.

Feinleib: I don't know why we didn't do it.

Bayles: I'm not sure what the nature of our agreement was with Mathematica, but it's conceivable that we were prohibited from offering a competing database. I don't know that for a fact.

Grad: Who would know that? Is that the kind of thing that Dick Glazer would be able to answer?

Rawlings: Or Dick Orenstein.

Starting NOMAD Development

Grad: Dick Orenstein. Okay, well, we'll pursue those with them. RAMIS, in those last two years, 1973-1974, you had renewed the license, is what someone said, but you had already started to work on NOMAD. Is that a correct statement?

Rawlings: Yes. Yes.

Bayles: I think we started, I mean, you guys correct me if I'm wrong, but all I remember was perhaps contributing something like writing a spec document.

Feinleib: Right, you worked with us on the spec.

Bayles: Right, I think that was around mid-1972.

Rawlings: Yes.

Bayles: Maybe till the end of 1972. I think I may actually have a copy of the spec buried somewhere in my storage unit. In my view, we essentially threw everything and the kitchen sink in. I mean, if we could conceive of building it, we could conceive of actually implementing it.

Boykin: At the beginning of this whole thing, when the decision was made to develop our own database system, the first thing we decided or discussed was, "Do we duplicate RAMIS or do we come out with a totally new product?" And there was that discussion. And we came to the decision that we wanted a better product, and we did not want to duplicate RAMIS, even though from a customer perspective it would've been much easier to transport their applications, if we had duplicated RAMIS.

Feinleib: I agree. It was a conscious decision.

Boykin: Yes, a conscious decision to make a new product.

Grad: And yet it was a similar kind of product. Is that a correct statement?

Feinleib: Yes. But much more--it had a bigger vision than RAMIS.

Pisarro: It was a much deeper product. The problem with RAMIS was that it had a lot of limitations to it. And, as NCSS, we couldn't get them to expand their vision as to what you could do with RAMIS.

Boykin: NOMAD was a relational database. It was the first commercial relational database that was out.

Feinleib: Well, I wouldn't quite call it relational. Not by today's standard. It was still hierarchical.

Grad: So we have a difference of opinion here.

Bayles: Right. I did a fair amount of programming – a couple of years of programming in SQL.

Grad: But SQL comes later, doesn't it?

Bayles: You have to understand that from 2000 to 2002, I did a lot of programming in SQL, which nobody would argue was not a relational database, I don't think. I can't tell you how many times I would start off a SQL statement with a LIST instead of a SELECT statement.

Rawlings: Because that's what it was.

Bayles: But the syntax was so incredibly similar.

Rawlings: It was.

Bayles: And it was 20 years earlier. Wondering whether the people who defined SQL ended up using us as a model. I don't know. I mean, changing the names to protect the innocent. But every time I sit down and do SQL programming, I don't do it anymore, but it just struck me that we were doing this in 1975! 25 years earlier!

Feinleib: Actually, NOMAD's language is sort of nicer. But SQL is more flexible than what NOMAD had.

Bayles: Well, yes and no, sort of. I mean, we talked about putting everything and the kitchen sink into NOMAD and it probably ended up being a 100-page spec document. We put in things that I didn't think we'd ever be able to implement. The one notable one being disjoint reports. This would be a nice thing to do, but I didn't know whether we could do it or not. Now it turns out that 25 years later, SQL still doesn't do it, and even an add-on, Crystal Reports, will not do it. It's what we used to call a disjoint report.

Rawlings: And we did it.

RAMIS Customer Base

Grad: I'm going to interrupt to ask one more set of questions, and then I'd like to switch over to NOMAD. Does anyone know how many customers NCSS had using RAMIS by the end of that time period?

Rawlings: 250.

Grad: Is that a made-up number, or is it pretty accurate?

Bayles: It's a totally made up number!

Rawlings: I'll bet it was pretty close.

Bayles: If somebody had asked me how many total customers CSS had in 1975, I couldn't answer.

Pisarro: I couldn't either.

Rawlings: What happened was that a bunch of us had left the operating system development group and gone over to form a product marketing group. The product marketing group had, among other responsibilities, tracking usage of the machines to try to figure out in some attributable way why people used our computer, and who used what. And to a certain extent I ended up tracking it more than others. So I had those numbers.

Grad: Was there much of a turnover on those customers that were using RAMIS, or did they tend to hang-in?

Rawlings: The latter.

Grad: They stayed with you. You didn't have a lot of losses year to year.

Boykin: They didn't have much choice.

Grad: The only alternative would've been to get a license from Mathematica, is that correct?

Bayles: I think so.

Other Third Party Products

Grad: One other short question that's related, you've mentioned COBOL Debug and FORTRAN Debug as being products that were running on your machines during that time period. Were there any other significant products running on the machines, on your own or third-party products, during that time period, 1970-1975?

Bayles: InfoTab was essentially our only financial modeling product.

Rawlings: We tried. We had a very nice [circuit board design] product called SciCards [from Scientific Calculation, Inc.] that Nick Pisarro used to develop the 3200.

Feinleib: What about the lens design program?

Rawlings: Well, there was a very good optical design program that was developed by Sergio Bernstein out of Perkin-Elmer.

Pisarro: But those were only used by P-E people, weren't they? They weren't offered for sale, or were they?

Rawlings: They were, and that was an issue, because there was some issue of ownership.

Feinleib: So who owned it?

Rawlings: Yes, that was a problem.

NOMAD Versus RAMIS

Grad: Given the time, I'd like to move over to NOMAD. Let's now go back to where you guys started with the decision to switch over. It had to be a concern of yours as to how you were going to migrate the users of RAMIS to your NOMAD platform. Who wants to start on that kind of discussion?

Bayles: I don't remember that being an issue early on at all, in terms of our design phase or the spec phase.

Boykin: I don't think that we worried about that, because we continued to offer RAMIS as we introduced NOMAD.

Grad: Did you continue to offer RAMIS after NOMAD was introduced?

Rawlings: Yes. And we had a huge amount of revenue continue from RAMIS for at least a year-and-a-half.

Grad: So people did not have to switch over to your new product?

Rawlings: Correct.

Grad: What was Mathematica's attitude toward that?

Rawlings: We had a disincentive for the sales folks to convert anybody. I mean, we wouldn't give them credit for it, basically.

Grad: What was the reason?

Rawlings: Because we wanted new revenue. We didn't want old revenue converted.

Grad: You didn't want to cannibalize your installed base?

Rawlings: Correct.

Grad: Does anyone know whether the agreement with Mathematica precluded you from offering a competitive product?

Rawlings: I think it did.

Bayles: I think it precluded us from offering a third-party competitive product.

Grad: Ah! Who was so smart to have written that agreement? Wow!

Bayles: Put it this way, in all of our development efforts-- and by the time this whole thing finished, I was in the finance operation and marginally involved with legal-- there was never an issue, either with Mathematica or from our own end that we were violating any agreement.

Grad: They didn't threaten to sue you or do anything of that sort?

Bayles: Correct.

Rawlings: I think they were busy with IBI [Information Builders Inc.] at the time.

Bayles: Yes, there was Mathematica and there was Gerry Cohen.

Grad: Nick, if you're willing to, can you tell us that story that you heard regarding Gerry?

Grad: Go ahead, Nick, if you want to tell that story now about what you had heard.

Rawlings: Well, what I'd heard was that Gerry wanted to be president of Mathematica, and Dick Cobb wanted to be president of Mathematica, and then Dick Cobb won, and so Gerry took off with his tape under his arm. And the next day Tymshare had a version of a database system that happened to be perfectly code-compatible with RAMIS, and you could move your application over, and they promised our customers that they would charge half of what NCSS was charging, and they could run it without any changes. They had no reference manuals, or training, so they used RAMIS reference manuals and training. That's what I understand.

Grad: Maybe there was some agreement with Mathematica?

Rawlings: No, I think Mathematica actually sued Gerry.

Bayles: That sounds familiar.

Grad: So if there was a suit, it's a matter of record, it's not a matter of opinion on that aspect of it.

Pisarro: I can add one thing there, though. I remember that when we came out with NOMAD, Gerry Cohen was completely taken off-guard. Both Mathematica and Gerry Cohen were completely amazed that this thing just came out of the woodwork.

Grad: They hadn't realized that you were working on it?

Feinleib: They had absolutely no idea whatsoever that we were working on this. It was kept pretty secret. We worked in a different location from the main office.

Pisarro: The development group, which was small, five people, six, maybe – the development group was housed in a separate building.

Feinleib: 441 Summer Street in downtown Stamford.

Pisarro: You were completely isolated from everybody else, which was probably fine by you anyway.

Feinleib: We had a great time. Hanging plants in the windows so people couldn't see in. And we hid the ping-pong table in the back.

Feinleib: And Judd and I would bring in these enormous pickles that he grew in his garden and put them out in the front, and watched people go by looking at them.

Bayles: That's an old joke about, "So, what are you selling? Why do you have that sewing machine in the window?" kind of thing.

Feinleib: People would knock on the door and say, "What are you selling in there?"

Rawlings: Yes, they thought we were a flower shop.

Pisarro: Because it was a storefront office.

Grad: Okay, we're now switching over to NOMAD, but just one more question on RAMIS. Once Gerry's IBI started to offer Focus services on Tymshare's system at half your price, what did that do to your RAMIS business? Who can answer that? Anybody?

Rawlings: Well, I tracked it, and it didn't have much of an impact.

Grad: Why not?

Rawlings: Well, it was difficult, I suppose, to know whether our lack of growth of RAMIS was because we were starting to push NOMAD, or because some of the customers converted to Focus. But I think Focus was slow to take off.

Grad: So that wasn't a major problem?

Rawlings: I don't think it had a big hurt to us.

NOMAD Development

Grad: Now let us switch to NOMAD. Who was in charge of the NOMAD development group?

Pisarro: Harold Feinleib.

Grad: Harold, please talk about how it got started.

Feinleib: I'm trying to think. I remember a conversation-- I forget who it was with-- probably Dick Orenstein, saying, "Okay, we have to build this product," and he asked me what I needed to do it. And I said, "Okay, I need a small group of people. I need a separate office, so that we're not distracted by all the goings-on in the company." And I went out and found 441 Summer Street, which we actually leased from a friend of mine, who I'm still very friendly with, who was the landlord. And we set up shop there. And there were one, two, three or four offices, some open area, and then we had a conference table, and a whiteboard or a blackboard, I forget which. And we sat around the table and we wrote stuff on the board, and we basically went through doing a lot of use cases, if you want to think of it that way, saying, "Okay, here's what we're modeling, here's a report we'd like to produce. Here's a data structure we'd like to model." And I remember us having a picture of an organization chart of a company, and we would constantly go back to that organization chart to design a report to say, sales or something by month or whatever. And it was a fun time! We had a great group.

Bayles: This is not gratuitous, but when I said you threw everything and the kitchen sink into the spec, I never imagined we could actually deliver all of it. We ended up delivering it and more.

Grad: Who were the people, Harold?

Feinleib: I read the book that was written by Daniel McCracken about NOMAD which was published in 1980. It references that NOMAD was started in 1973, but I'm not 100 percent clear. I know that Judd Boykin was on the team, Dick Bayles was there in the beginning while we were writing the specifications, and he did most of the writing of that.

Bayles: Mike Bayuk.

Feinleib: Nick Pisarro, Mike Bayuk. Nick Rawlings, were you part of the initial group?

Rawlings: I think I joined a year after it started. Then I stayed. I was the Director of Product Marketing at the time. What happened was you guys were all in Product Marketing, and Nick Pisarro worked for me. And you were only going to borrow him for a while. I think Dick Bayles talked me out of Nick for a little while. He said, "I'm only going to borrow him for a while."

Boykin: I have another thing that I think is important. At the time the decision was made for NCSS to write their own database management system, Orenstein said, "You have two years. You have to have the product ready when we renegotiate with Mathematica." And we actually did it in a year-and-a-half, didn't we? And we then started beta testing before the two years even came about. Isn't that correct, Harold?

Feinleib: Yes. Actually, the agreement that I remember is that I said, "Look, I will tell you when we're one year before release." That was so that the rest of the company could start to get prepared. And I picked that date, and then delivered to that date. And I think it was about two years, Judd. It was maybe a little less than two years when we came out with NOMAD. Well, I know that we got into beta test prior to when we said we would be ready. I know we were ahead of schedule.

Boykin: It was ahead of schedule, that's true.

Rawlings: There was one other guy.

Rawlings: Jim Steuart.

Boykin: He came in a little later, though. NOMAD was actually released, or nearly released when he came on.

Rawlings: I don’t think so. I think he was pretty early on.

Boykin: He wasn’t in the initial team, but he did come on later. I don’t know if it was after or before release.

Grad: So there were a half-dozen people. You were sort of a skunk-works operation.

Rawlings: Absolutely skunk.

Grad: One question. Did any of the people on that team have internal knowledge of the RAMIS program?

Boykin: No. We didn’t want internal knowledge of the RAMIS program.

Grad: You really set up a clean room, but you didn’t have a problem that way, because apparently you hadn’t been on the inside of the code. Is that what you’re telling me?

Feinleib: Correct. And in fact, we did not know RAMIS like the back of our hand, either. We knew a little bit about it, but not a ton.

Boykin: We didn’t *want* to know, because it had so many problems.

Rawlings: A different thing was that it was written in FORTRAN, and we wrote the whole NOMAD in Assembler.

Feinleib: Yes. You have to understand this was a little before the days that C was really popular.

Boykin: Oh, yes.

Feinleib: And before there was really a C compiler on a 360.

Pisarro: And FORTRAN was probably the worst language you could write something like this in.

Feinleib: That's true.

Charging for NOMAD Use

Boykin: But there was a reason for using Assembler. Remember, in time-sharing, efficiency was extremely important. And writing in Assembler gave us the best opportunity to make this the most efficient product we could.

Rawlings: One of the ways in which efficiency was important is that we switched from charging by the CPU second, to charging sort of whatever we wanted.

Feinleib: Gronks, we called them Gronks.

Rawlings: Yes, yes.

Pisarro: Really? We did that?

Rawlings: That was my invention. It was after the BC comic strip.

Pisarro: Who else would come up with Gronk?

Rawlings: But no, it was the ARU, the Application Resource Unit.

Feinleib: That's right.

Rawlings: And we had a Gronk table, and the Gronk table had the product factors of how much to surcharge CPU time or I/Os. And we could, in a sense, turn up the factor if we made NOMAD more efficient.

Feinleib: Very good. Good memory!

Rawlings: So what we were able to do was sell a gallon of gasoline for a higher price, because people could get more miles to the gallon.

Grad: So in some sense you were charging for the result or what was produced rather than how much time it took you to produce it.

Rawlings: Yes.

Feinleib: Correct.

Rawlings: Or how much resources it used.

Grad: That's very significant, because IBM was often considered to be interested in slowing its machines down, because it made more money. I used to say that we didn't write IBM software to run slowly on purpose, but they didn't tell me not to.

Feinleib: You cynic, you!

Pisarro: Well, that's why CP/CMS was kept under the table by IBM so long was because it ran their machine so much better. So I'm sure that had some affect on it. You know, it was also a little backdoor bit of software.

Grad: It was backdoor. We couldn't get the product division to pick it up. I had it in DPD, instead of in the product division. Let's go back to what you were doing here with the ARU; it was a means by which you could charge for value rather than charging for time

Rawlings: Yes.

Grad: That's a great idea. Did you ever do that with RAMIS?

Boykin: No, I don't think so; no.

Rawlings: But what we did do was we did surcharge for RAMIS. We surcharged it by 25 or 30 percent.

Feinleib: On a CPU time-basis.

Rawlings: Right.

Grad: Okay, so you just see those charges and you say, "Okay, you're going to pay this much more for those CPU minutes than you will for the others."

Rawlings: Well, people sort of said, "I'll run this report, and how much will it cost?" "It'll cost you a hundred bucks." And if they had run it by writing their own program using exactly the same resources, they might have been able to get it for $75. But because they used RAMIS, they paid $100. And what we did was we turned around and gave that $25 to Mathematica.

Grad: That was the uplift that you paid back to them, so you got full value on your time.

Bayles: In the RAMIS transaction, that was probably true. Actually in most of our third-party contracts, that was what we were focusing on. We were focusing on not eroding our raw time charges.

Rawlings: We wanted to get a surcharge.

Bayles: Anything you could charge over that-- at least early on-- anything you could charge over that went back to the vendor.

Grad: So in other words, your revenue was your time, and the excess revenue, you were, in effect, paying as a royalty to the vendor. Is that a correct statement?

Rawlings: That was the model.

Feinleib: That was the model probably for the first five or seven years of CSS.

Grad: So now with NOMAD, you have a whole new ballgame.

Rawlings: Well, we are our own vendor. So we sort of wanted to get paid for NOMAD.

Grad: Did you separate in your internal accounting between the time-sharing usage of NOMAD, and the "what you were charging the customer?" Does anybody know?

Rawlings: We tracked it.

Pisarro: I should know the answer, but I don't.

Rawlings: I do know the answer. We did track it. We tracked it in a hundred different ways. Basically, we wanted to know how much of it was surcharge, how much of it was what we called direct non-surcharged. And we also attributed additional revenue. So that if somebody was editing a NOMAD file, or editing a RAMIS file, we could say, "This revenue that we got from the editor was because they were a NOMAD or a RAMIS user. So we were able to come up with a way of figuring out where our revenues came from.

Delivering NOMAD

Grad: Let's look at NOMAD. Now it's early 1975. When do you release NOMAD to the marketplace?

Rawlings: First usage was March, 1975. First release was October of 1975.

Grad: Why the six-month lag?

Rawlings: Because we had some early Beta testers.

Grad: So this was an extended testing period of almost six months?

Boykin: Yes.

Grad: Who was involved in that?

Rawlings: Was Ira Solomon?

Feinleib: From UNUM Insurance near Boston?

Grad: Was the same team supporting the product, and making the bug corrections during that period?

Rawlings: Oh, yes.

Feinleib: In fact, there was a lot of development after the product was released. We were certainly making lots of development improvements.

Rawlings: We didn't stop developing it for another ten years.

Grad: Let's look at that six-month period. So from March to October, you were not getting revenue from the product yet?

Rawlings: We were.

Feinleib: Well, what about the night when we released it, where we came in with sheets?

Rawlings: That was October.

Grad: Tell about the naming of NOMAD. I gather that's a good story.

Feinleib: I think Dick Bayles named it. It was named after "NCSS's Owned and Maintained Advanced Data Management" System.

Bayles: Yeah, it was something weird like that.

Pisarro: Two points for you, Harold!

Boykin: It was supposed to be a name that we would use as a development name, and we would come out with a marketing name at a later point. It just stuck.

Rawlings: There was also a notion that the people who initially worked on it -- Hal and Judd and Mike -- had been working on a couple other sundry things before they launched into this. So maybe they were the nomads. And then there was a guy, a good guy, John Skodon, who called us the Gonads.

Feinleib: He lives down in Tucson, Arizona. He still does work for us at Aperture.

Rawlings: He was probably one of the first 20 employees. Maybe 30.

Grad: Tell me about the sheets when you released the product in October?

Rawlings: Well, we had a silly sort of show that the marketing group, I think, Jim McGuire put on.

Feinleib: Right, Jim McGuire was the marketing manager for NOMAD. And we had decided what we would do is we'd set a target for the first month's revenue, and I think it was $40,000 or something like that. And he actually went to Brinks and got $40,000 in cash, and put it on a table, and all the developers wanted to look like nomads...

Rawlings: Like Arab sheiks.

Feinleib: Sheiks. And we came into the room, and that was during the launch of NOMAD. And it was sort of funny, and we actually exceeded that initial month's revenue considerably over what we had sat as the target.

Bayles: That was done at the hotel at JFK, right?

Rawlings: Yes.

Bayles: I flew in from the West Coast for that. I remember that.

Pisarro: They brought in all the salesmen; this was the big product introduction internally, so they flew all the salesmen in.

Bayles: Right.

Pisarro: And I don't know if they had training or not, but they flew everyone in for this big product introduction. And I still have the sheets we wore!

Grad: You're kidding me!

Pisarro: We sewed up the holes and we've been using them for the last 30 years!

Bayles: You're not telling me you're using the sheets!

Pisarro: Well, we actually used them if we went out camping or something; we used them around the Christmas tree to, you know, catch the needles and things like that.

Grad: So you're still getting multiple usages out of those sheets.

Pisarro: That's right.

Grad: That's a wonderful story.

Bayles: Now, how much money was that first month's revenue target? Does anybody remember exactly? Was it $40,000?

Feinleib: I thought the money we brought in from Brinks was more than that.

Rawlings: I remember $100,000. But maybe that was what we hit. Maybe that wasn't our target.

Feinleib: Yes, I think the target was like $40,000 and we did $100,000. It was very, very good.

Rawlings: Very dramatic, yes.

Grad: Were there trained professional services people, Judd, by that point in time or not?

Boykin: Well, at that time I was not part of professional services.

Grad: Who was in charge? Do you remember?

Boykin: I don't remember.

Grad: Alan Connor?

Rawlings: No, no. Alan Connor was CSS.

Feinleib: What about Mike Field?

Rawlings: No, Mike Field was trying to sell the operating systems.

Bayles: The other possibility was Joe Grispo.

Rawlings: I think that's – I think it was Joe. That sounds right.

Grad: Okay. And before the October announcement, were there people trained within the company to help customers use NOMAD or not?

Rawlings: Yes.

Feinleib: We trained the support engineers that worked in the field.

Rawlings: They were called tech reps, technical representatives.

Feinleib: And that was led by Larry Smith.

Rawlings: And Ruth Morley?

Grad: Now they were trained to do what? To help customers use it, or to teach them how to use it? What was the training?

Feinleib: Both.

Grad: Who prepared the training materials?

Bayles: I have a feeling Larry did a lot of that.

Rawlings: Larry and Ruth, I think.

Feinleib: I think one of the things I did was write the original NOMAD manual.

Grad: Had you written the stuff for the RAMIS manuals, because you didn't like their manuals, too? Or was that someone else that did that?

Feinleib: No, someone else did that. I only write manuals for my own software.

Grad: I see. Incidentally, Harold, I've got a copy of "Line by Line," that you had recommended. But I haven't started reading it yet, but I will do so.

Feinleib: I never read past the third chapter, but the first three chapters tell you how to write. And that was quite an eye-opener.

Advantages of NOMAD

Grad: Harold claims that his outstanding writing style is all due to "Line by Line." That's why I mentioned it. Okay, you have made the announcement. Why did it take off? Because people had been using RAMIS, they were happy with it. What were the special things about NOMAD that made it take off so effectively?

Pisarro: There were some things in NOMAD that really made it a 4GL, because NOMAD had a language; RAMIS did not have a language. RAMIS wasn't really a language. The only way you could interact with it was by typing commands in, which is sort of a language. But NOMAD had a whole language. You could actually write whole interactive programs in this language to do things.

Feinleib: And Nick Pisarro wrote that piece of the product, that language. And you could do substitutions, you could run a report. You can take the output of that report, and re-sort it, re-group it, things like that.

Rawlings: And the procedural language that Nick wrote also had go-to, and it had labels, and it had variables, and branching, and all sorts of other good stuff in it.

Grad: So it was a full-range procedural language.

Rawlings: It was a full procedural language.

Pisarro: RAMIS did not have a language. That's why I made my first statement, that NOMAD was really the first 4GL, because it was the first database language where you could develop whole applications in it, and we got customers that wanted to do that.

Rawlings: I guess the point might be that it had third generation language components that made it possible to do things. RAMIS or NOMAD could claim that with a 4GL that a single statement, like List or Select or Report or Load, you could do a whole bunch of stuff. And that was why it was considered fourth generation. [The disagreement between me and Nick P was that NOMAD was categorized with 4GL's, but the power that Nick P talked of was in the 3GL elements - the NOMAD procedural language, which he wrote. He was rightly proud of his invention, but it was 3, not 4 GL. One of my favorite examples of how powerful the language was, was cut from James Martin's book on *Application Development Without Programmers*, 1982, Prentice Hall. He found it embarrassed the other products too badly.]

Boykin: One of the real plusses that people saw was that you could get multiple reports with one pass of the database. Whereas with RAMIS, you had to pass the entire database for each report.

Grad: So it was far more efficient in that regard.

Boykin: Much more.

Feinleib: Judd wrote the report writer, I believe. Is that correct, Judd?

Boykin: I wrote the report writer, but Nick Pisarro wrote the processor to give me the information that I needed.

Feinleib: So you did the formatting of the report. The formatted output.

Boykin: Correct. And I never got into his code and never understood it. Not ever! It was incredible.

Rawlings: Well, I think the schema was a huge part, and Mike Bayuk wrote a big part of that.

Boykin: Yes.

Rawlings: And the flexibility with schema check, and schema re-org, which is what I wrote.

Grad: Was this compilable?

Boykin: Later on.

Rawlings: Essentially not.

Feinleib: No, it's still interpretive, but it did hunks of operations. So the things that it did were big, like run a whole report with one statement.

Boykin: Correct.

Pisarro: Right, you didn't work line by line, you specified a whole operation, like gather all this data, and sort it in such-and-such, and order, and combine it with this data, and then output it with one statement.

Bayles: I wouldn't underestimate the influence of the schema re-org, and the ability to modify, within limits, the structure of the database on the fly without having to redo everything all over again.

Pisarro: Right. That was a biggie. That's true.

Feinleib: Did I write any code?

Rawlings: Yes, you wrote the SORT part of LIST. And you also wrote the early EVAL [Expansion Evaluator].

Bayles: Harold's memory's like mine. He and I don't remember much.

Feinleib: Now that you mention it, I do remember fiddling around with the Sort program. That's funny.

Boykin: Well, you know what's amazing, I nearly didn't come on to this phone call, because I said, "I don't remember anything." But in talking about it, everything's coming back. There's so much that comes back; they're good memories.

Feinleib: That's what I said. I didn't think that any one of us, or I didn't think I could remember very much of it, so that's why we formed this phone call.

NOMAD Maintenance and Enhancement

Grad: And now NOMAD is announced and it's starting to sell well. You guys are doing terrific. So, let's talk more about NOMAD. You're continuing to develop and operating it on your time-sharing system. Did you now move back into the main shop? Or did you keep it there at 441?

Rawlings: Mostly we kept it there.

Feinleib: Yes, we stayed. One of the things that we did, which I thought was sort of brilliant, is that we had a terminal set up in 441, and anytime NOMAD hit a bug, it would write it out to that terminal. So that we, as developers, would see all the users using it on the time-sharing system, and when we saw a bug, we'd go in and fix the bug.

Pisarro: It'd be fixed by the time the person reported the bug.

Boykin: We'd say, "Well, try it again." He'd try it again; it wouldn't be there!

Feinleib: Right. And this was really funny, but it happened all the time, that we would fix the bug, we'd put up a new version, and then by the time the customer would call in, support would say, "Oh, I think we've just fixed that bug. Why don't you try it?" We had a version called X-NOMAD, and they just were able to launch X-NOMAD, and lo and behold, their bug was fixed!

Grad: Wow!

Feinleib: Is that good service?

Grad: That's incredible!

Boykin: Well, the whole community was on one computer in those days. It was sort of like the Internet.

Bayles: It's like an ASP service today.

Rawlings: Oh, better.

Grad: So you could see what was going on.

Feinleib: Absolutely.

Grad: That's impressive. Let's take it up to 1978-1979, did you at any point then move the NOMAD group back into your main operational location?

Feinleib: Yes. In 1976 or so, we moved it to the upstairs of 485 Summer Street.

Rawlings: We didn't move it back to the headquarters, which was in Westport.

Boykin: Well, we did eventually.

Feinleib: Eventually we did. But we all moved to Westport together.

Rawlings: Correct.

Feinleib: There was another development group, the VP/CSS development group, which was at 500 Summer Street, across the street. And we moved to 485, which was upstairs from our computer room--I think the computer was still there at that time. And Doug Grant joined the group at that time. I remember Judd and I went on vacation to St. Martin at that time.

Boykin: Correct.

Rawlings: And Bob Jesurum joined then, too.

Feinleib: He worked on NOMAD?

The GIST Project

Rawlings: Well, no, he worked on the planning for the next version.

Feinleib: Oh, GIST. That's like a bad dream. I forgot all about that.

Rawlings: Harold and I stopped talking for two years.

Boykin: Larry Smith's big talk about GIST. I'll never forget that.

Grad: Was that a follow-on to NOMAD, or was that something different?

Feinleib: The idea of GIST was to integrate a number of applications. Now, there have been companies that have been very successful. I think their name starts with an M, of integrating a suite of applications under one banner.

Bayles: Ends with a T?

Feinleib: And ends with a T, right. And there's an S in there! So I think that was the claim to fame of Microsoft of having the Microsoft suite with interoperable applications. At that time, the applications we were using were quite primitive. But the idea of GIST was to have an overall integration – one environment where people would be able to share data among applications.

Rawlings: Right.

Grad: Now this time period is when that GIST was being thought about?

Rawlings: Oh, 1978-1979.

Feinleib: Maybe 1978.

Grad: Was this before the acquisition by D&B?

Rawlings: Yes.

Grad: But it didn't interfere or stop it?

Feinleib: I think it may have caused the acquisition.

Grad: There's a story here obviously. Was this a lot of money? Was this a lot of conflict? What was going on?

Rawlings: Well, it was a classic boondoggle software project that somebody got going. That's what it was.

Feinleib: It never got very far. There was almost no money put into it. Besides Bob Jezurum, who else was on that team?

Rawlings: There were a couple of long-haired guys.

Feinleib: In fact, the Russian guy [Alexander Stepanov].

Rawlings: But I think that's where I ended up getting exiled to work with Dick Floeting for two years, because of it.

Grad: Can you tell us the story?

Rawlings: Well, my view is that I said, "This is bullshit."

Grad: To whom did you say that?

Rawlings: To Harold.

Grad: And Harold was not happy with your comment?

Rawlings: I think not.

Grad: And Harold, do you remember this?

Feinleib: Oh, yes.

Boykin: I agreed with Nick Rawlings, except I kept my mouth shut.

Grad: So you didn't get exiled to Anchorage, Alaska or something.

Boykin: Correct.

Feinleib: That was a part of my life that I don't want to remember.

Grad: Now, Harold, as I understand, at some point in time, you started to work on a different assignment, you were working on something else.

Boykin: GIST.

Grad: Was that GIST that you were working on?

Feinleib: I guess.

Boykin: Yes, remember they split development. It was under you. I took over development, and you took over research. And that was GIST.

Grad: Now this has got to be what? 1977-1978?

Rawlings: Oh, 1978, maybe early 1979.

Feinleib: Remember we moved to the new headquarters about 1978.

Bayles: That was 1978.

Feinleib: In the building that Dick Bayles designed.

Rawlings: Yes, you had it built.

Bayles: I was the building guy.

Feinleib: Dick Bayles had the most varied career at National CSS of anybody. From finance to legal to developer, to building coordinator, building designer, machine room designer, too.

Bayles: Yes, that's true.

Grad: He never could find anything he could do well? Was that the problem?

Rawlings: No, he did everything well.

Bayles: Jack of all trades, and master of none. That's why.

Feinleib: Did you open up the California office, too?

Bayles: Yes, I built that building and the machine room there, too.

Feinleib: It really is incredible how many different roles you've played, Dick. It was terrific.

Bayles: Yes, but none of them very confidently.

Rawlings: Do I get to blame you for "the Banies?" The guys from Banes?

Bayles: No, no, you don't get to blame me for that! [D&B brought in a crew of MBA's from Harvard. The company name was Banes. We called their employees "the Banies." It's my view that their advice to jettison small clients was wrong, and that when we followed that advice, it almost killed us as a company. Dick was in planning at the time, and he participated in the analysis - as did I. Neither of us agreed with their conclusions; however, we were overruled.]

Rawlings: All right, good.

Bayles: We're out of school here, but I'm still on your side on that operation.

National CSS Management

Grad: Help me here. NOMAD's working, you're selling it well, the company is going well, and at some point there, 1975 or so, there was a change in management.

Rawlings: Right.

Grad: And this is when Bob Weissman becomes president. I'm not going to pursue that process at this point, but Bob becomes the new president?

Feinleib: Bob Weissman didn't happen until after we moved to Wilton.

Bayles: No! Not true!

Feinleib: No, he was president in Westport.

Bayles: It happened when I was there; I know because his office was next to mine, next to the conference room or the boardroom. It was probably 1975 or 1976.

Feinleib: Right. It happened during the NOMAD time, because I remember that I started with Dick Orenstein, I believe, and finished up with Bob Weissman.

Bayles: Correct.

Boykin: Because I remember Weissman coming over and talking to us.

Rawlings: Yes.

Boykin: Giving a good pep talk.

Feinleib: You have to understand that the company had a lot of sanity to it, but there were veins of insanity. Just a few.

Bayles: You wouldn't be talking about Providence, would you?

Feinleib: Well, that was one.

Boykin: Well, you know, I remember the famous story of IBM threatening to pull the machine out if they let Bob Bernard go into the machine room again. Because he always thought he could fix everything, if it would go down.

Grad: Nick Rawlings, this sounds like an early story, I gather. Is that correct?

Rawlings: Sure.

Grad: When Weissman comes on, does that change your strategy with NOMAD?

Feinleib: That's not correct at all.

Bayles: Bob for the first couple of years didn't really change anything.

Rawlings: He didn't know anything.

Feinleib: Well, the thing he did know was he knew how to sell the company.

Rawlings: Yes.

Feinleib: And he understood that fine.

Rawlings: Well, he knew how to accept the offer from the guys who tried to buy Tymshare.

Bayles: But that was four years later; in the first two years, he didn't try to do anything. I mean, I shouldn't say that. He ran the company.

Rawlings: He tried to keep some of the big guys--Bob Degan and Dave Fehr and Alan Rievman and John Sterling, from pulling each others' hair out.

Bayles: I had lunch with Bob Degan and Bill Bloom about three weeks ago.

Grad: Would they be interested in participating in this in terms of the history of National CSS?

Bayles: Well, actually, I talked to Bob Degan about it. The answer is yes. He has an interesting perspective.

Boykin: He does.

Bayles: At least in the early years. I mean, he left the end of 1975.

Grad: Okay. But 1970 to 1975 period is certainly an interesting period in the company's history.

Rawlings: Yes.

Pisarro: Bob Degan was the operations manager of the company, and he really put that into shape in terms of running the data centers.

Feinleib: He did an excellent job.

Rawlings: He sure did.

The NOMAD Story Continued

Grad: So from 1975 to 1980 staying with our NOMAD discussion, NOMAD continues to be developed and worked on. It continues to be marketed, and becomes a more and more significant part of the business, or sort of stabilizes?

Bayles: More and more significant.

Feinleib: Yes, I remember right up until the acquisition [by Dun & Bradstreet] we continued to enhance NOMAD. We had some very large customers like Bank of America. They're the ones who said, "Look, we have to have statistics." That became a big project. And it was extremely well accepted at the time. But it was done mainly for Bank of America. Also then we had a project where we converted NOMAD to run under VM.

Rawlings: And that was also for Bank of America. They gave us a million bucks to do that.

Feinleib: That's right.

Rawlings: And we said, “We’ll pay you back when we sell it.”

Feinleib: Correct.

Grad: And did you sell it?

Rawlings: But I think that was more like 1981 or 1983 or something like that.

Grad: One of the things I’m trying to get to is that when the company was sold, was the revenue and profitability of NOMAD a major factor in that sale?

Bayles: Yes.

Rawlings: I think so.

Feinleib: NOMAD was a big factor in that sale.

Bayles: It was the other big market for our product, because software development was really tapering off. There were then other ways of doing that.

Feinleib: Yes, VM became big in the marketplace, and so big companies could do their interactive development that way. And NOMAD became certainly the NCSS flagship product. It was generating the bulk of the business.

Boykin: Okay, now if I remember correctly, and you can correct me if I’m wrong, one of the reasons that Dun & Bradstreet wanted a time-sharing company was as a method of delivering their products. Their products being market analysis, etcetera, etcetera.

Rawlings: That was one reason. I think another one was that they were feeling that they had fallen behind, that TRW had outpaced them technologically, that they were a really ancient company. And in addition to maybe getting time-sharing and NOMAD, they also got an incredible network group who knew what they were doing. And they really needed that.

Grad: Did you have your own network system by that time?

Rawlings: We had a very extensive network, and packet switching, and we knew what to do, and we had computers all over the world, and we were interconnecting them.

Feinleib: That was another major project. NCSS competed with Tymnet.

Bayles: And we were doing packet switching as early as what? 1976, I think.

Feinleib: Yes. That's when I think we started doing packet switching.

Bayles: Right, and then we built it up into a substantial enterprise, which became DunsNet and DunsNet, for all I know, is still running.

Rawlings: So I think that whether they bought us for that reason, or not, that was a major asset that they brought in and took advantage of.

Bayles: Maybe the only asset that's left.

Interpretation versus Compilation

Grad: Now let's talk about NOMAD once more. Did it stay interpretive for its entire life, or did it ever become a compiling product?

Feinleib: You'd have to ask Nick Rawlings. Nick, you stuck around working on NOMAD for quite a while, didn't you?

Rawlings: Yes.

Feinleib: We did compile it at one point. Didn't we, Nick?

Pisarro: Well, we compiled the byte code, it was sort of run like the way you run a JAVA script program today. You compile it at the byte code level.

Rawlings: You got pseudo code, and you're running the pseudo code; you're not running the source code. And Nick Pisarro came up with a pre-scan thing so you could save the scanning. You save the "pops" and you execute them. So that was an important piece, the pre-scan. But basically it stayed as an interpreter all the way through. The expression evaluation was almost all interpretive. And so it continued to be slow in that area.

Grad: Let me ask a quick question here. So in some sense you did not compete directly with the database management systems, with software ag, with IDMS and so forth. Or did you feel you were competing?

Feinleib: Those are all compiled systems, if I remember correctly.

Rawlings: Well, I think we competed. We didn't compete with them maybe for different reasons, and that had to do with the fact that we didn't run on MVS for the longest time. We didn't run on native VM for the longest time. And those systems ran on MVS and DOS.

Bayles: And I also think, Burt, that your definition of compiled may not be what Nick and I and Harold and Judd would think of. Compiled, to me at least, means you're running native machine code.

Rawlings: But if you have a statement that says, "I=I+1" somewhere you've got a single instruction that says, "add one to 1."

Bayles: As opposed to a pseudo code, or a meta language, or whatever the heck you want to call it, which is something half-way in-between the two where you're parsing.

Rawlings: [In FORTRAN or NOMAD, you can have a source statement I=I+1. When that statement is executed, the result is that the variable I is one larger than it was before the statement was executed. The issue of compiler vs. interpreter is most glaring with such a simple statement. A good compiler would turn that into just a few machine statements, e.g.

```
LOAD Reg7 with I
ADD 1 to Reg7
STORE Reg7 into I
```

An optimizing FORTRAN compiler would note where and how the variable I is used and might decide to keep its value in one of the machine registers. The code it would generate would be a fast, single instruction, e.g.

```
ADD 1 to Reg6
```

In an interpretive system, such as NOMAD and most early BASIC systems, the source is first scanned and an object code, of sorts, is generated. The "object" code left after scanning is a set of programmed operators or "pops", where each pop is a subroutine call. When execution of the program was under way, an interpreter would see not I=I+1, but

```
evaluate I+1, assign to I,
```

where "assign" and "evaluate" are subroutines that get called during execution. Those subroutines might take hundreds or thousands of machine instructions to do their jobs. The single source statement, then, takes thousands of instructions to execute.]

Grad: And that's a very efficient way to do it.

Bayles: And you've turned it into running much more efficiently, but it is still interpretive.

Grad: Okay, so the degree of interpretation is the point that you're making, and I think that's good.

Bayles: My point is that, if I look at the AS/400, which is an SQL-based machine, I don't think that the SQL procedures are compiled in the sense that we talk about compilation. They're certainly not run by interpreting the character string. They're run with a somewhat codified version of that. But it's not native machine language.

Grad: One last question on NOMAD, then I promised we'd be done at 6:00.

Rawlings: Well, we could do this again. It's fun talking to these guys.

Feinleib: It's only 3:00 in California.

Grad: I keep forgetting that. Back to NOMAD. Did you do many sales of it directly to users to run on their own machines, either from a time-sharing, or regular operations basis?

Boykin: Toward the end.

Rawlings: After 1985, that was sort of our only business.

NOMAD on the PC

Feinleib: Nick Rawlings, why don't you talk about NOMAD on a PC?

Rawlings: One of the things that happened was that PCs came out. And we had to figure out what to do with NOMAD there. Now what Gerry Cohen did with Focus was that he recompiled it, because it was written in FORTRAN. And here we were stuck with a million lines of Assembler code. And there was no way we could run it. So we set up a new group. Harold knows about this better than I probably. But we set up a new group, and they came up with essentially a new design. Data Dictionary-centric, and it was going to solve a lot of problems that they saw with NOMAD. And other than being called NOMAD, and having some character strings that were similar, it was going to be quite different.

Bayles: Really? I didn't know that. That's interesting.

Pisarro: Did that actually get developed? Because I left in 1980.

Rawlings: I have a shrink-wrapped copy of NOMAD for a PC.

Feinleib: And I know Bob Schraer when he worked at Aperture, he used to use NOMAD a lot back in 1990 - 1993.

Rawlings: Well, what happened was that that project went for a couple years and got nowhere. And we spent all the money. And then what we did was we retracted, and we tried a couple other methods. We bought a company that had a little product, and we thought we could rewrite it, to add some NOMAD features, and we tried that for another year. And then we tried a third attempt, and we wrote NOMAD in C and it was very similar to the Mainframe product. Still, it didn't have all that many features. It didn't have hierarchical structures- it only had relational, only single level, lots of flat tables. And then we had a lot of problems, because we had to make it would work with Presentation Manager as well as with X-Windows, and with DOS, and then with Windows, this weird thing that Microsoft was putting out. And so we wrote it in this convertible compatible fashion that never ran. And then we also had to make that work with DEC-VAX and with some UNIX machines. And so we got a couple teams to make that happen. And we actually did sell some to people. Basically, they bought the mainframe product with the promise that the C-NOMAD, as we called it, would work. And it never really did. And even to this day, I think that there are some successful examples in Motorola, and maybe the State of Illinois, where they got some decent applications running. But by and large, it's generating zero revenue today. The only thing that's being used is the Mainframe product.

Closing the NOMAD Story

Grad: Let me try and draw this to a close. Tell me, each of you, when you stopped working with or related to the NOMAD product. Harold, start with you.

Feinleib: Oh, God. I think I was involved with it until I left NCSS, which was 1980.

Grad: Dick Bayles.

Bayles: Involved with, as opposed to using.

Grad: Yes.

Bayles: Well, I left in 1981. But actually involved in the development side, was probably 1973.

Grad: Nick Pisarro?

Pisarro: Oh, when we moved to the Wilton office. So that must be 1977.

Bayles: I think it was 1978.

Grad: Okay, Judd Boykin?

Boykin: I think that NOMAD remained under me as far as development until I left, which was about a year after we were purchased.

Grad: So that's about 1980.

Boykin: Yes.

Grad: And Nick Rawlings, I guess you were the last man on the ship. Is that correct?

Rawlings: Well, the ship is still sort of floating. It might be awash, but it's still floating.

Grad: You told me there's still revenue coming in each year.

Rawlings: Yes, there's probably $5 million in revenue every year.

Bayles: Are you still there?

Rawlings: No, I'm not still there. I left like eight years ago to go to Yale. But I was still writing code in 1997 for NOMAD, and I continue to consult on NOMAD. I did some consulting on NOMAD last year.

Grad: Who owns it today?

Rawlings: The larger company is called Gores. And the smaller company underneath it is called Select Business Systems. Also known as Select BS. Which I think is great! That actually is their logo, "It's Select BS!" Come on, guys! You know, pay attention! Pay attention. It's like the guys who had this, I don't know, telephone system that they call Fluke. You know, "If it works, it's a Fluke!"

Other NCSS Products

Grad: That's wonderful. All right, we're going to bring this to a close. I had asked whether – besides the wonderful experience with GIST – there were other products that were being built by National CSS during the 1970s.

Bayles: The only one I can think of is our own version of Basic.

Rawlings: We did develop Basic, but there was another one, it was I-Spice. Remember that, guys?

Bayles: I-Spice. The IC circuit designer?

Rawlings: Right.

Rawlings: We actually hired physicists from IBM, there was a guy from India, there were a bunch of others. And there was this big uproar when some guy took off with a tape, and ended up getting arrested in his house. Remember that?

Boykin: Yes.

Grad: That sounds like a story I'd like to have on our website.

Boykin: Well, I think he worked for me.

Rawlings: I think you're right, Judd.

Boykin: Those are different stories.

Feinleib: There was one about the NCSS 3200.

Rawlings: Oh, yes! That was an important one.

Grad: That's a separate story. Alan Rievman spoke to me about that a bit.

Rawlings: Well, Alan Rievman knows the financing, which was an interesting story, but Nick Pisarro sort of thought up the 3200 as kind of an interesting dream when playing with SCICARS and some other stuff.

Pisarro: Oh, no, I didn't have anything to do with it.

Rawlings: I gave you credit, Nick.

Pisarro: Oh, thank you. I'll take it, but I didn't do it.

Rawlings: Was it Skodon who came up the 360 on a chip? On a card?

Feinleib: We hired an outside company to do that.

Pisarro: Well, but that's not the NCSS 3200, that was done by a different company. I had nothing to do with the NCSS 3200.

Bayles: In conjunction with a company in, say, Oakland or Berkeley, one or the other, or maybe it was South Bay. I'm not sure.

Grad: That'll be another story that we'll pursue.

Bayles: The 3200's another whole story actually.

Rawlings: And to a certain extent, the 3200 was pushed with NOMAD. There was this, "You can't buy the hardware without the software. It's a bundle." It was a big push on NOMAD with, "You buy NOMAD, and VP/CSS, and the 3200, one big thing."

Boykin: Let me just say one thing. You keep asking about other products, et cetera. In reality, there were three products that really made NCSS. The FORTRAN debugger, the COBOL debugger and NOMAD. Those are the ones developed by NCSS that basically made the company. Does everybody agree with that?

Rawlings: Yes.

Feinleib: I would say, yes.

Bayles: And the basic operating system itself.

All: Yes, absolutely.

Grad: I was going to say, VP/CSS was your foundation.

Rawlings: Yes.

Feinleib: Correct.

Grad: And you have to admit, though, that RAMIS was certainly significant, and led you to NOMAD.

Feinleib: Definitely.

Rawlings: It led us from a marketing standpoint. We became aware that there was a market.

Grad: That's right. And you therefore developed an understanding of that market opportunity, and then built a far better product.

Feinleib: It was the germ of the idea, also. It was more than just the market. It was the germ. You know, it gave us the idea of what could be done.

Boykin: But I was relating to the products that we ourselves developed.

Grad: As against those from third parties. I see. You did your own FORTRAN debug, and your own COBOL debug?

Boykin: Yes.

Grad: Gentlemen, let me thank you. You were very well disciplined for a bunch of undisciplined guys.

Boykin: It was enjoyable.

Grad: Did you all enjoy it?

Bayles: I absolutely enjoyed it. Very nice.

Grad: Okay, I do ask you now in closing the thing, please, do go to the website, do read what's there. Harold has written a terrific article there about the early days. Nick Rawlings has just added a whole bunch of good stories. We have Dick Orenstein's remembrance of the first five years. There's good stuff there. And I hope you now will be motivated to add some of your own stories. If you run into any troubles in using this system, please give me a call, or drop me an email, and I'll try and assist you. And thank you very much! And you'll get to see the tapes as soon as we transcribe them. Okay?

Bayles: Thanks, Burt!

Pisarro: Okay. I have a whole basement full of stuff.

Grad: Don't throw anything away! Doug Jerger will talk with you about collecting. Don't throw anything away, please.

[Prior to conducting the conference call, a set of questions had been prepared and provided to the participants. These questions are reproduced below]

Subjects for Discussion

One of the significant early NCSS business decisions was to license to sell time-sharing Services using RAMIS from Mathematica. Please provide further information about this decision and its results:

1. When did this happen? What motivated the decision?

2. What were the terms of the license? Who negotiated the license on both sides?

3. Could NCSS modify RAMIS or did they have to ask Mathematica to do so?

4. What kinds of technical issues did NCSS run into in using RAMIS on a Time-sharing basis rather than for regular batch operation? What technical changes did NCSS make in the language or the system? Were these NCSS' property or Mathematica's?

5. Who were the key technical people at NCSS who worked with the RAMIS program?

6. Was NCSS's focus on using its Time-sharing environment for RAMIS primarily to develop the applications or to run the applications after they were developed? Was it primarily intended to be used for RAMIS debugging (like COBOL DeBug and Fortran DeBug) or was it expected to be mostly for operational use?

7. Did customers have to get a license from Mathematica in order to run the programs developed on their own systems after using RAMIS in the NCSS Timesharing system to construct and debug the programs?

8. How many customers did NCSS have for RAMIS? What industries and applications did they represent? What was the annual growth rate and the loss ratio? Were the customers actually using NCSS Time-sharing for debugging and testing or were they also using it for operations?

9. Was there a substantial professional services operation in place in order to help customers write programs using RAMIS or to write them for the customers?

10. What were the reasons for discontinuing the RAMIS license? When was this done? Who made the decision?

A second key decision by NCSS was to produce NOMAD, their own 4GL language and system, to replace the use of RAMIS. Please provide further information about this decision and its results:

1. Was the decision to produce NOMAD made primarily for business or technical reasons? Who made the decision? What were the principal factors? What were identified as the main risks?

2. How did NCSS expect to avoid intellectual property issues with Mathematica?

3. How was the new product specified and by whom?

4. Who was on the development team and what roles did each play? What was the development schedule and how well was it adhered to? What were the principal technical problems and how were they solved?

5. What was the market that NCSS expected to serve with NOMAD? Was it different from the RAMIS market?

6. Were there any other competitors at that time selling Time-sharing services using a 4GL? If so, what companies and what products were they using?

7. How were the internal applications people trained on using NOMAD? How were customers trained to use NOMAD?

8. Where did the NOMAD name come from? Who came up with it?

9. What professional services were in place to either produce NOMAD programs for customers or to help them write these programs?

10. What were the business results from the NOMAD services? Separate the processing services revenues from the professional services revenues? Was this a profitable business?

11. Did NCSS license NOMAD for use by its customers? If so, under what terms and conditions? Was it licensed to other processing services firms? If so, under what terms and conditions?

12. What happened to the RAMIS business? Was NCSS able to migrate its customers from RAMIS to NOMAD? Did this cause any problems with Mathematica?

13. What did NCSS view as the principal advantages and disadvantages of NOMAD versus RAMIS and Focus?

14. How long did NOMAD remain a viable product in the marketplace? Did NCSS migrate the product to other platforms?