THIS IS THE BEGINNNG OF AN INTERVIEW WITH BUTLER LAMPSON AT CAMBRIDGE RESEARCH LABS].

INT: When did you start working at DEC, with DEC, around DEC?

BL: 1984(?). I was working in research. Bob Taylor and I joined DEC to start SIC.

INT: Why was there a need to start SIC?

BL: From our point of view, it happened because Xerox decided to fire Bob Taylor. So that caused Bob Taylor Brown for someplace else to go and to take his group. DEC was the obvious possibility. It was a company that was big enough to be able to do research. There aren't too many of those. DEC hadn't really had a research lab in the classic, basic research sense. It had a thing called, - Eastern Research Laboratory. That was much more of an advanced development laboratry and tended to take much shorter term things and much tighter hooks into natural product development. I think, on the whole, it was generally recognized that as a research lab it wasn't very successful and that if the company was serious abouat having a research lab, you would have to start with more of a clean slate and try to set things up specifically with that in mind with a long term charter and keep it fairly well separated organizationally from the development organization. I think Sam saw this is a golden opportunity to acquire a very talented group of people. It's not so easy to build these labs; it's not easy to find a really good person to run on; it's not easy to establish the nucleus of senior and talented people. So that's what happened.

INT: How did you originally hook up with Bob?

BL: I've known Bob since the late sixties when he was head of the Information Process and Techniques Office at ARPA(?) and I was a graduate student and assistant professor at Berkeley workig on a ARPA funded project. Then, in 1971 I had been at a start-up _____ computer corporation which folded and Bob had just joined Nason(?), Xerox, Palo Alto, research labs, had a charter to set up a computer laboratory, computer research laboratory. So he was scouting around. He had drawn several people for CSS Colton(?) [INAUDIBLE]. So that was how I originally started working for him, that was in 1971 and I've been working for him ever since.

INT: What was the beginning of that environment like?

BL: At Park?

INT: Yeah.

BL: Well a lot of people have written things about that. I've read [INAUDIBLE].

INT: From your observations, perspective?

BL: Well, I think .... from my perspective the way Park ___ got started was, the part of Park that we cared about got started was the CEO of Xerox made a pronouncement that Xerox was going to invent the architectured information, of course that didn't mean too much, but what did seem to mean a lot was that it was pretty clear to people like and we thought it was also pretty clear to them that if Xerox wanted to continue to be a growth company in the long term, they would have to more than build copiers. The obvious thing for them to do was go into electronic ______ processing and the automated office. It seemed to us that here was a company that had a real need for the sort of work we wanted to do, and had enough money to be able to afford it. In a sense, didn't really have the option to decide after two or three or five years that they weren't interested in this any more. So we went to work for them with the idea that we would invent this architecture information, we would show how to make automated offices - and we did. From the visionary's point of view the whole thing was completely obvious. The basic notion of the work that was done in _____ was to say, well there's all this well established technology: A) interactive computers have been aroud for 10 or 15 years and B) producing images on pieces of paper, that technology has been around for hundreds of years at least and it's not(?) going to be feasible to put those two things together.

INT: What portion of that field were you interested in?

BL: I was interested in all aspects of it. I worked on all aspects of it. I'd be interested in various bits and pieces of that for a long time. [INAUDIBLE]. In the sevenites it was just becoming possible to do this in a much more serious way. So we sat down and tried to solve all of the aspects of that ______ we thought we could get our hands on. So we built work stations and ____ displays and we built rapid software and screen editing software and we built local networks so these things could continue to share information. We built ____ file sytems and we built laser printers so you can the images that were produced and actually print them. We sat down and systematically built all the pieces of that environment. We were fortunate that the technology to do most of those things was just becoming available. We were also very fortunate that Xerox had people who could handle the electromechanical and optical parts of the printing because we didn't have the talent for that. We were able to _______ very effectively and Gary Striker(?) did and in effect already had all those ideas well under way but didn't quite know how to drive them, he knew how to build in such a way that we could actually get the images that we wanted. That was the part that we knew how to ______.

INT: Then in '84 when your team came over, how was that different? What elements did you bring? What elements did you find that there was in Digital - pressures(?)?

BL: By that time I had been at _______ for near fifteen years and we pretty much realized that initial vision and were in a phase, what you might think of as, consolidation and extension. I think it's pretty fair to say in general that the whole of the eighties for the whole ______ industry was a period of consolidation and extension. It's hard to point to a single, dramatic new idea that was thought up and prototyped in the eighties. Pretty much what people did was to polish and commercialize the ideas that had been thought up in the sixties and seventies. And this whole business with personal computers and work stations and local networks and plants that were computing and all that, - that was all invented in the seventies. And I can't point to anything that was invented in the eighties that is similarly waiting for exploitation now. We didn't, either at Park or at Surf(?) invent anything comparable in impact to the things that were invented in the seventies. But what we did do was work on trying to make it all a lot more, work on a much larger scale and work much more reliably and work much more predictably in the sense that you could try to build tools that would make it possible for people who were not wizards to do these things that previously had required wizards. Then also we tried to extend things in the obvious directions Ethernet is good, then something faster must be better. If work stations are good, multi processor work stations must be better. So we worked on that. That was pretty much the pattern. In many respects it was much better being at DEC than being at Xerox because, after all, that was an engineering company and it's full of extremely talented, competent, energetic engineers and it's very responsive to engineering ideas. Whereas, the first approximation, Xerox is a marketing company. It's not full of lots of talented engineers and the ones they do have, tend to have expertise in toner physics and things like that, not thinks of tremendous interest to us.

INT: Digital seems to have an engineering history of extending existing ideas in scale and liability.

BL: On the other hand, I think in many respects, working at DEC has been very frustrating because we came to DEC with the idea that we understood a lot about work stations and distributed systems and this was a great opportunity for the company to pick our brains and use all this knowledge that we were gifted with to make spectacular progress. I would say, they didn't do that, which is too bad.

INT: That seems to be very muc like DEC, isn't it?

BL: I think the fundamental reason is that the VAX-VMS time sharing was so spectacularly successful that, first approximation, people didn't see any need to do something different. So that is why, I think, on the whole, the whole development of work stations and distributed computing at DEC has been stunted because the other stuff is so successful. So, you've got a combination of, first of all, making a bundle of money doing this other stuff. So, why should we do something new? Second of all, the fact that you have an enormous installed base and a lot of momentum, it is extremely difficult to deflect. And thirdly, a strong element of NIH. It's very clear from the perspective of 1991 that the company really blew it both in personal computers and the work stations, pissed away very large sums of money and given up large growth markets that the company from a technical and historical point of view, should have been able to own but choose not to. And that was frustrating for people who came to DEC with a lot of knowledge that was directly applicable to those things. And the reaction to that in the company was, - oh yes, obviously these things are strategically important, you guys know a lot about it but somehow we're not going to really do anything.

INT: But there's some part of that culture that is supportive.

BL: Oh yeah, there are lots of individual successes but I think some real opportunities were missed, which is too bad.

INT: Talk about that area that you found at DEC that goes between advanced development and privatization. How do you see that working?

BL: I think generally the experience has been that there are two ways to do it: one way is, which is not a way that we practice to general, the recent interpretation of practice to general, is that you can build something as an AD project and then you can wave a magic wand over it, declare it a product. And that was done, for example, with the Ethernet bridge.

INT: Once the prototype was going?

BL: Right. This is really great and this is really pretty close to being a product and we'll just ship it. Obviously, I'm over-simplefying but fundamentally that's one paradigm. The other paradigm is, you do some research or some advanced development and you develop a prototype and you learn about yourself and then you take everything you've learned and you start a product development. You don't attempt to use any of the actual lines of code or pieces of logic design from the prototype. That has some obvious advantages and some obvious drawbacks. The advantages are that, you get to design the product with the benefit of both all the knowledge that we gained from the research and advanced development phase and also the benefit of all the knowledge that we have about how products ought to be designed. Whereas if you do the thing the other way, it's unavoidable. There are a whole set of compromises that are going to have to be made. Chances are that the AD people don't really know how to design products. But they're going to have to worry about that a fair amount if there's going to be any chance that their AD is actually making it possible to make it a product. So that has a lot of difficulties but it has the advantage that if you can see that you obviously can get the job done much more quickly. In general I think the research labs have followed the other paradigm. So, just to pick a random example, for instance, we did a lot of work on ______ ______ of threads(?) and built a couple of prototype systems that used threads fairly heavily and then a couple of guys from the research organization and some guys from the research organization and some guys from Spitbrook work together to develop a specification and a design for a product version of the [INAUDIBLE].

INT: And at the same time into productization.

BL: And at the same time into productization, exactly. But they didn't use any of the code that was _______.

INT: Again, I've heard this back and forth, the new model is to do quick prototypes and then throw them away and do real products. Perhaps, because of the politics and phase review, product is decided upon the standard Digital way does seem more to be - put something in advance development [UNINTELLIGIBLE]. Is that a political restrictionn that prevents the second prototyping model to be more used wisely in civil engineering(?)?

BL: Build the AD prototype and ship it?

INT: No, that seems to be the old way.

BL: Well, that's not always the way it works with AD projects.

INT: You said it works different. Sometimes it's better, sometimes it doesn't make any difference.

BL: I think .... that approach works well when you have some .... you need something that's quite isolated in order to make that work, something like the Ethernet bridge, is a good example because it's spec is completely defined by the Ethernet stack, especially in the early days when people didn't worry too much about how you were going to manage it and things like that. You could go out and build that thing with the idea in mind that you would be able to walk up to some product, even in installation, plugs the cable in and if you did your job properly, it would just work. You didn't have to negotiate a lot of interfaces with other people or worry a great deal about how this was going to fit into the grand scheme and things. And in that environment you can have a group that goes off, does a prototype of that thing and then assuming that you can CSSE to accept it, you're in fairly good shape. It's much, much more difficult to do that with something like the threads example where, it has to fit into a rather complicated environment and you have to negotiate with standards bodies about things and so on and so forth. So, there's bound to be a lot of development and evolution and in that environment the value of the initial implementation, first of all the value of the initial implementation is a lot less 'cause things change as you go along and it's very difficult to build initial implementation, paying attention to all the constraints that the products all through development have. The other thing there is when the functionality of the thing is new, you need to expect to do more experiments to find out what's right and what's wrong. So, it's not so realistic to believe that the prototype that you first build is going to be something you even want to ship.

INT: So it depends on how advanced the research is, how advanced the development is versus how advanced the standards are?

BL: I think it depends more on the extent that you can really isolate the thing that you do from the rest of the environment, very cleanly. And, what that typically means is that it will work when there is some existing interface defined and you have a great idea about how to implement some aspect of it better. But it's not going to work so well when you're inventing a new piece of functionality.

INT: What are some of the other projects that have been going on in research labs that have both turned into advanced ____ projects and have not?

BL: There's a number of interesting examples. One of them was a project that we had at SRC (Systems Research Center) ~~Sirk(?)~~ to build a very high performance multi processor out of high tech ECL gate arrays. We did a lot of design work on that. Eventually we abandoned the research project because the prospective gate array supplier decided that he couldn't really make the gate arrays. But, after Argonaut was cancelled, the Argonaut design team picked up this whole design approach and invented a VAX implementation called, Raven, based on this philosophy, this strategy for how to build a high performance processor out of high density echo gate arrays. And they found a different supplier, Fuditso(?) which seemed more credible than the supplier that SRC ~~Sirk~~(?) had. And that actually was a [UNINTELLIGIBLE] .... for some considerable distance and unfortunately ended up failing for the same reason, namely that the gate array supplier couldn't really deliver. But that was a fairly successful example I think, of transferring that piece of technology, that approach to high performance processor design from the reseach organization to the development organization. If the gate array supplier had made it, then it would have been a very competitive product.

INT: Were there any sillier examples in which development took on something that wasn't so big, an idea that actually started working on development that proved to be impossible?

BL: I don't know, maybe they're _____, this was one example of that. I'm not sure I can think of any other examples like that. Usually, the difficulty is that the development organizations tend to be quite conservative so, if anything, you can't persuade them to stretch far enough as opposed to persuading them to stretch too far. The reason this Raven thing happened was that Argonaut had a very conservative design and it got cancelled because it was too conservative and therefore it just didn't perform well enough. The result of that was the group was very much inclined to stretch because they observed that by not stretching, they got their machine cancelled. Unfortunately, it didn't work out any better the second time but it certainly was a much better thing to do in the sense that it at least had a chance of working out really well, whereas, the very conservative thing was bound to be overtaken by events. But usually our experience has been that the development organizations are pretty conservative. Our difficulty usually is to persuade them to sign up for something, what we think, there's no doubt that it can be executed successfully as opposed to having to worry that they're not going to take on something - sorry - as opposed to having to worry that they'll take on something that will turn out to be too hard for them.

INT: What is the motive of working in the research labs? Where do ideas come from? How do you guess what direction to push in?

BL: Well, you try to look forward and figure out, what's going to be important. You try to figure out what's going to be important for the company both by thinking hard about it yourself and by going around and talking to people throughout the company, trying to understand what they think is going to be important. And then, you try to find some match between that and some talents that you have or some ideas that you have for how to attack the problems. That approach is the basis for most of the projects that are undertaken [INAUDIBLE].

INT: [INAUDIBLE].

BL: Those are applied mathematicians. They're style of work tends to be very different because they don't undertake big projects. Usually there's only one person or maybe two people working on something and it only takes a few weeks or months to carry out any particular enterprise. So, for example, Andrea Brodeur did some work on figuring out how to use - the standard technique for doing table look-up and mapping some key into some value is called, a hash table. And the problem with hash tables is that in the worst case they can perform badly, on the average you expect that they're going to perform well. So if you're in a situation where you can't afford to have the worst case be bad, you can't really use an ordinary has table because you can't guarantee that the worst case isn't going to be bad. And what Andrea was to work on, is a gate by which essentially you could catmate(?) several hash tables together and thus gain enough additional flexibility that you could guarantee ..... has tables are primertized(?) by a hashing motion which tells you how to compute from the key, where you should start looking for the value. And normally you keep the hashing function fixed. The unfortunate thing about that is, you try to make the hashing function be a fairly random function of the key but if you're unlikely in your choise of random function, you'll find that in a particular situation, you have a lot of keys hashed to the same value and then it will be slow. So the obvious thing to do in that situation is to change the hashing function. And what Andrea did was to work out systematic scheme for doing that and was able to prove that every time you change the hashing function, you'll again have a high probability that things will be good. And, if it still isn't good, well you can change it one more time.

And then the question is, - how to put all that together so you can characterize its overall behavior precisely. [INAUDIBLE]. And then those ideas were transferred to the MAC organization where they've been using them to build a high performance bridge. So that was an example _____ _____ on a much smaller scale. So, much of what we've been saying about projects and so on isn't really applicable ______.

INT: Outside of that were there a number of people working on ______? How do you manage that research as a project of just a day to day of work?

BL: Typically what happens is that somebody or some set of people get some idea and work out a proposal for something that might be done. Then in some fairly informal way, that proposal competes with other things that might get done. Usually in the research labs, the way that competition takes its form is that, in order for you to do your project, you have to be able to persuade people that they want to work on it as opposed to working on something that they thought up or something that somebody else thought up. In addition to that there's some amount of, loosely speaking, top down judgment by the senior people and the lab managers and senior engineers from other parts of the company about whether projects are a good thing or not. So, by some combination of this bottom up, people voting with their feet, and some top down judgment about what's, A), what's promising and B), what's important. You reach an agreement that a particular project is going to be undertaken by these people and then they work on it.

INT: Can you give an example of bottom up and top down approach?

BL: We always have to have both because it doesn't work for the management or even the senior researchers to say, - this is really important, we got to do this. If you don't have any good ideas, you know, people really don't like to sit around and just scratch their heads and brainstorm week after week and month after month. It doesn't really work. So you always have to have a thing working from both directions.

INT: Can you give an example of a project where somebody really felt ...

[END OF SIDE A].

INT: ... just started pushing the product in some direction?

BL: Well, I can tell you the story of a particular sequence which illustrates both of these things ___ ____. I've been several years working on high speed switch in local area networks and built something called the Autonet(?) which you might have heard about and Bill Cartney(?) working on a second generation one. The first generation Autonet was actually started at my suggestion. I said, - [INAUDIBLE]. Why don't we do this other thing instead? And I actually proposed doing it in a particular way and this attracted the attention of several people. They started to work on it and after some months of work decided that what I proposed was actually not the right thing to do. [INAUDIBLE] ... similar in flavor but significantly different in detail. That then turned into the Autonet 1 project. Then, more recently, we've been extending that, trying to figure out how to make it run at higher speeds and added functionality. Again, I think that was motivated originally from top down, where I sit, and [INAUDIBLE]. The original Autonet was very successful but there's not really any hope of getting people to adopt something different from MBI(?), that runs only on the megabits(?). [UNINTELLIGIBLE] ... we already have a standard there, we don't need this things. So you ought to target this technology at the next higher speed. We started in on that but then - and the other thing that we said was, - well, we ought to, in the beginnig think about how to get this _____ standard. So what are the real needs going to be and also what are the standards organizations committee is going to insist on. So we said, - it's going to be important to be able to handle essentially field implement fixed bandwith circuits in this network as well as the standard kind of packett switching stuff because for applications like shipping video around, they're going to want that. That was sort of a high level view of it. Then, from the bottom we had one of the folks working, - Mat Gogan, who felt that it was really important to work on congestion control. So he put a lot of effort into that, worked pretty closely with _____ Jane(?) and some of the other folks in MAC who had been working on congestion control in wider inputs. As a result of that, Auto net II has adopted a particular achictecture for how to ship the data around in small chunks called cells and how to do congestion control for those cells which was not something that we had contemplated at all when we laid down the original goals for the enterprise.

INT: It sound like there's a lot of synergy with development.

BL: Well we try to do that. Sometimes it works well, sometimes it doesn't work so well. But generally the research enterprise goes much better if you can work closely with people in the development organization.

They have a lot of power so they also have a lot of knowledge about what the problems are on their side of the fence. Eventuallay the research has to be accepted by them or it's not going to be much good for the company. We can still write some nice papers but revenue and profit are not going to increase.

INT: The question would be, - how much and what kind of buying(?) you try to?

BL: Right and that varies a great deal. We always try for as much as possible but also our view is that fundamentally we're being paid to make these judgments and we're going to make them the way that we think is best regardless of whether the development engineers agree because we're not paid to just do what the ____ engineers think is important. For that you just need ____ organizations, you don't need a research lab. One of the important functions of the research lab is to exercise independent judgment on what's going to be important in the future. But in general things work better when you don't have these disagreements so we prefer to have it that way.

INT: I'm amazed to find you so responsible when God knows what you're going to find wandering the halls here, people who are working on elephant ____.

BL: No, that's not a responsible way to spend the company's money and in the end it's bound to backfire. We feel that it's an important part of the researcher's job to think in an appropriate way about what's needed. But that way is very different from the way in which an engineering development manager works. Also, you want the researcher to think in a way that won't cause anything worng, reasonably and frequently, otherwise he's not being sufficiently ________. You're not so comfortable with the notion of engineering development project is going to fail. That happens to, you expect that to happen sometimes but you don't have much of a tolerance for it because a lot more is riding on it.

INT: How do you run fail safe? How do you give people the freedom to fail? Can you give an example?

BL: The NT project that I mentioned before was an example of something that definitely failed. It was recognized from the beginning that that was a risk and we said, - this is a good thing and it's worth doing. If the supplier falls throught, well, too bad. We'll still learn a bunch and furthermore this is a risk that's worth taking. And everybody felt very good about that. In fact, some people thought, Stewart for instance, said, this was the best possible outcome because it's actually comparable. If Sirk(?) had gone ahead and built this new imcompatible machine out of this crackpot technology and the company got everything out of the project that it could have gotten if the thing had been carried through to completion but it was much, much cheaper. So, from that point of view, it didn't fail at all. But from other points of view, yes, it did fail in the sense that there was a definite goal to build this machine and the goal was definitely not achieved.

INT: I'm interested in you dealing very often way beyond the realm in which there are standards, in which case you tend to derive standards. What are some of those areas?

BL: Well threads is one example that I cited where we actually, people in the research organization worked very hard to push the thing actually through the ______ process. Usually we don't do that because the other parts of engineering have people that are experts at that. It's our preference and ______ their preference too, to let them do what they're good at. But in this particular case was a situation where we had some special expertise and special talents that we were able to apply to that task. I guess we're hoping to be able to do the same thing with the second generation Autonet, it's still fairly premature. The noise level is pretty high.

INT: The XDI seemed to be easier, came out easier?

BL: NO, I think XDI was extremely painful actually. What is true is, that there weren't a lot of competing proposals. And I think the reason for that was that when Ethernet came out, it was well in advance of what was really needed, so it had a pretty long run. And for quite a while nobody was really, only one or two people were really thinking seriously about what a next generation thing might be. Then by the time more people started to pay attention it, somehow this F~~T~~DDI crock had gotten enough momentum that everybody said, - well, oh and the other thing I guess is that people hadn't really been doing any AD so there weren't any competing things that could be demonstrated. And so when people suddenly woke up and said, - we're going to need this inter(?) megabit thing, there were no competing things that were working. Of course the XDI(?) thing wasn't working either but it had the virtue of having had the standards wheels turn four or five times, so it had the momentum and the only way you could have overcome that momentus was by coming in with something else that actually had been prototyped. But nobody had built a prototype of anything else that was at all competitive.

INT: Where the next generation.

BL: Quite a few people are building prototypes of this and that.

INT: Because they can see an immediate need with imaging.

BL: Exactly, so it's going to be much more complicated and confused. But we can hope that we'll get a better result. The ~~E~~PDI is really a crock. It's disgusting.

INT: As long as the _______ hit the wave. I want to go back to the number of people in group dynamics that works best. How does a researcher team work, the same ways in which engineering teams work but in a researcher image(?), what's a team, project people, what's the role of the individual, the role of the team? Where does creativity, innovation come back from?

BL: Well of course that varies a lot from one enterprise to another. Generally we try not to run projects that require more than, say, four people because then they get to be too difficult to manage and the management overhead gets to be too large. Typicall the researchers aren't that good at management. Also, when you have a big team and you need more management, that tends to stifle creativities. And for all those reasons we try to avoid that. When you have a small group you don't need to have everything so well defined as you do when the group is larger. The group can meet three times a week and ideas and can be passed back and forth very freely and that's typically the way it does. Usually, we try to do the standard thing of taking the horn and dividing it up into chunks and each guy gets to be the expert on one particular part but there's a sufficiently small number of people that usually everyone can know a lot about what's going on in the whole project. There are lots of cross fertilization and changes in roles and so forth, that's usually the way it goes. The issues of how you manage a team development tends not to play as big a role as they do in a development organization where necessarily the ____ are bigger.

INT: What do you think maybe the ____ ______, most effective research team has been? I'm looking for interesting examples.

BL: I think there's been a number, there was the group, I guess it was three people who built the original _____ ____ on the [INAUDIBLE]. That was transferred more or less directly into product development, there was a thing called the fire box which also didn't work out all that spectacularly well for a variety of complicated reasons, mainly having to do with the fact that VAX got overtaken by RISC as processors but certainly the original fire

____ was extremely successful and it was a small number of people got the thing done quite quickly. Then there was group of, I guess three or four people who built the basic software for the fire fighter. [INAUDIBLE].

INT: Was that working with the Dragon Chip II?

BL: No, Dragon Chip was a graphics chip. [INAUDIBLE].

INT: Was there any connection with say, the chip development?

BL: No in that case the chip was done. It had already been done by a completely separate group at Hudson.

INT: Talk about cross fertilizations between those groups. What about in addition to the academic community and other research organizations, is there any?

BL: Oh yeah. There's of course a whole community of researchers who publish in journals and go to conferences and all that sort of thing. DEC research in general is certainly an active participant. If you look at the program for almost any conference, you'll find people in [INAUDIBLE] serve on the programs and publishing papers. We run all the labs fairly active, ____ scientist programs where people come from univeristies and spend three months or six months or a year ____ in the lab. [INAUDIBLE] ... or Brandeis. Several years ago, [INAUDIBLE]. Often those give rise to very active research collaborations once the people go back to their own institutions.

INT: And in those research collaborations, what kind of example _____? Obviously, there's always been relationships I know of _____ ____.

BL: So, for example, Jim Horning(?) and John ______ and MIT have been working on _____ program specification for a number of years and that's recently given rise to a ____TLE or DEC SPEC. They've been adapting these techniques to direct specifications or C(?) programs and built tools to support this process. Now, that was just started a year ago and now the second year we're going to work on focused responsible _______ to try to use these methods to write specifications for ___________. That work has been a very active collaboration between Jim and John and now folks at TLE and one or two of John's students and associates. And, in fact, there's actually someone from CNU(?) development as well. Antother example that I mentioned, Ed ______ from the University of Washington has spent some time, about five years ago I guess, actually worked on [INAUDIBLE] implementation and then went back to the University of Washington. They acquired some Fireflys(?) and they did some outstanding work on [INAUDIBLE] ... much faster. But certainly, it's established a state of the art for how well you can do and then that form of inter process communication fell fast. That was all done by building on the Firefly hardware and software which we exported to them. Another thing that's pretty important is that all the labs run summer student prorams. You typically have anywhere between five and fifteen summer students from universities all around the country, sometimes all around the world. Often they do interesting research when they're there for the summer and also we end up hiring those people when they get their degrees.

INT: Any interesting ideas from there?

BL: Quite frequently, yeah. All kinds of different things happen in summer school.

INT: Are innovative researchers primarily 20 year-olds, then afater 30 you become a manager, start getting washed up?

BL: No, I don't think that's the pattern that we generally see. I think, one of the most ____. Thirty is a little young to be washed up even in the most active fields. Most of these guys just haven't been around that long. I've been in this field for thirty years myself, which is almost as long as anyone has been in and I'm only 47. Maybe that will happen in another ten years but it hasn't happened much yet. I think we have very few people who started out as researchers and then turned into managers.

INT: I've seen some evolution of the clever engineers of the sixties tending tending to turn into like managers and then turning, in turn, to like service reps or something. Is there a waterfall slowdown from research, researchers go into engineering? What happens? Do old researchers ever die?

BL: I don't know. There's been a little bit of flow of researchers into the development organizations but it hasn't been dramatic. At Park there were lots of people who left research and joined other companies [TALKING TOGETHER - UNINTELLIGIBLE]. Often, astounders and backed away(?). For example, well astounders are pretty high level managers. For instance, Eric Schmidt left Park and was one of the founders of Sun and is now President of their software company or one of their ________. Chuck Guest and John Warnock left Park and started Adobe. Bob Metcalfe left, actually the first went to the Xerox development organization and then left and started 3 Com(?), then _____ left Park and started, Convergent

Technologies. There were six or eight other ______ like that. We haven't seen that so much from the DEC labs perhaps because they're not old enough yet.

INT: The movement from the seventies of innovation to the extension and scaling of those technologies and bringing them, easier to use, users for the eighties. How does that vision move from eighties into the future?

BL: Well that's a good question. There's still a lot of opportunity to continue this process of scaling and extention, moving to higher band widths and to making things work on a larger scale and work more reliably. I think you can see, at least another decade, of productive work along those lines. It's an interesting question, whether there's going to be anything really dramatically new in the nineties. I have a long term vision of where computing is going. I like to tell the story by organizing it around three themes, each one has a slogan that goes with it, one of them is the idea of having direct access through your computer to a large quantity of information. So the slogan for that is - the Library of Congress behind your screen and that's a _____ very far away from today in the sense that the limit of what you can do today is to get the network news behind your screen. It's clear that there's no fundamental technical obstacle to getting the Library of Congress behind your screen but there's a lot of problems that will have to be solved. And I think, when that's achieved, as I'm sure it will be sometime within the next five to twenty years, I think that will result in a really qualitative difference in the way you vote in view of the utility of the computers. That's been a dream for a long time but I think it's fair to say that so far not much progress has been made in actually achieving that dream.

INT: [INAUDIBLE].

BL: I don't think we're that much better off than we were in the late seventies ______ achieving that goal. But it will be very surprising to me if ten years from now, I think we will be a lot farther along, ten years from now towards achieving that goal. I think it's pretty likely that there will be some dramatic changes resulting from that. My second theme is the notion that you can use the computer to model something else, model something from the real world but maybe not _____ mathematics and whatever. And the slogan for that is, - virtual reality. There are other sub-slogans like - scientific visualization and so forth. I think the thing that's going to make that qualitatively different from what we see now is that we're going to get to the point where the machine actually can present you with a sufficiently convincing represenation of what it's models are, that it feels real. And that's something we can just barely see happening in a few areas now. Some aspects of mechanical CAD for instance, just, just barely begin to see that but it's still really high tech stuff. Boeing is building a 777 for example, they're putting a lot of effort into this and it's their hope that they're going to be able to, by computer modeling, figure out, when the pipe is going to collide with the electrical cable and actually see that. But that's something they can just barely do by dint of investing a huge amount of effort. And it seems very clear that 20 years from now, for instance, if you're going to build a building, you would be expect to be able to walk through the building, when it exists only inside the computer, see it in the same way that you will see it when it's real and have all the details of how it goes together - properly modeled and available for inspection. That's a combination of a lot of database work, so the machine actually has all the necessary information, a lot of graphic work so that it can actually present it and get it through your eyeballs effectively. In many cases I think a lot of modeling work, and it's not so clear for the building, but many other applications you're going to have to have in the machine, some models of the physics and chemistry of whatever it is going on in the real world so that you can compute. All of that stuff is sort of arbitrarily hard depending on how far you want to go. I think it will be very interesting to see how it evolves. Some things will probably go much faster than others. So, for example, I think the modeling for building the modeling part is relative easy, the database part is actually fairly difficult. The graphics part is reasonably difficult. If you want do fluid flow, the modeling part is very hard, the database part is pretty trivial and the graphics part is, I think, more or less under control. So it varies a great deal from one area to another. So that something, over the next ten or fifteen years, we can expect to see some quality of changes in what the mission can do. But that subject certainly won't be exhausted ten or fifteen years from now where it's conceivable the Library of Congress would be nearing completion, ten or fifteen years from now. That obviously won't be true with virtual reality because there is so much that we can imagine doing. I don't know if you've read a science fiction novel called, "Greed ____", where the idea is that instead of having archaic games the computer can generate holograms well enough so you can make a fantasy world which you can enter and play in. If you think about how much computing is required to achieve that, we certainly won't have that 20 years from now. That's something where the possibilities are almost limitless. Then the third area is robots. My slogan for that is, - I want to have a robot house maid ~~made~~ and I'm convinced that that's doable. But again, that's something that people have dreamed about, ______ progress is slow. I don't know as much about that as I know about the other areas. I know at MIT the I(?) lab, they now have a robot which can ride a unicycle. It's not completely satisfactory because it has to have a cable attached to it. Someone has to run around behind it making sure that it doesn't trip over its own cable. I've actually seen it work.

INT: Having tried to unicycle through Harvard Yard several times, I [INAUDIBLE].

BL: I've only seen this thing operating on the steps of 545 Peck Square. And because of the need for the stupid cable I don't think they're quite at the point where you could ride it through Harvard Yard but maybe in a few more years, that [UNINTELLIGIBLE].

INT: That looks like an interesting vision in trend and movement.

Bl: The trick in these things if you're trying to run a research lab is to find something that's easy enough, you can actually make a sizeable dent in it. My current guess is that probably the one of these things that will yield most rapidly is the Library of Congress. I'm actually have been doing some thinking about that. It's much more difficult, maybe it's just that I'm older and grumpier, when I joined Park in 1971 I thought it was pretty obvious and people have written ____ _____ books ____ future, how amazing it was. I really thought it was pretty obvious. It seemed to me like a low risk enterprise. There was not really any doubt tahat it was going to work. Nobody else wanted to do it.

[END OF SIDE 2 -- BEGIN SIDE 3]

BL: ... so everybody wants to work on these things and that was definitely true in the seventies, so its much more difficult. I think it's going to be difficult to do spectacularly better. But things that are going to be spectacularly successful are likely to be the things that are obvious and then they're obvious to you, they'll probably be obvious to somebody else too, maybe not to everybody but to some of the people. I don't know, it's a puzzle.

INT: I love your, either spectacularly better dramatically new. That's interesting.

BL: If you think like the way things go in physics for instance, every now and then somebody does something that is really unexpected; high temperature super conductivity was really unexpected. My perception is that doesn't happen in computing, everything is more obvious.

INT: Has DEC since you've been here following the obvious?

BL: DEC has been behind the obvious for the most part, in my opinion. That's one of the sources of our current problems is historically I think DEC has been, historically I think DEC has been right up there doing the obvious thing. In the seventies, time sharing was the obvious thing. We did it and did it really well. Networking was the obvious thing too and it was obvious in the sense that, in my judgment anyway, there was really no doubt that it was going to work. It's true it was new, you had to open new markets, customers were going to have to do things in new ways in order for you to make money. But it was obvious that it was a good thing. Maybe you couldn't convince the business school graduate but anybody with the slightest trace of imagination, it was obvious it was going to be a win. I think if you went and talked to Ken, that's exactly what he would tell you. In the eighties, the thing that was obvious was that work stations were opening. I thought that was obvious in the early seventies but certainly by 1980 there was no question about it. The next problem I think was that by that time the other stuff had been so successful that it was much more difficult to see that a new thing was obvious.

INT: Blinded by success.

BL: Blinded by success, precisely. I think that has been a real problem with DEC during the eighties and it continues to be a problem.

INT: What is DEC's psychic role in contribution to computer ______, to advancing the state of computer technologies, advancing the ______. What's the niche?

BL: My perception is that the niche for DEC is to build what customers and applications builders will think of as bay(?) systems on which they build their stuff and to make those systems cost effective and reliable and scale(?). There's a whole range of different technologies that goes into doing that everything from magnetic storage to networking to hyper processesors(?) to databases and so on. Of course one of the things you observe in this bay systems game is that the definition of a bay system is constantly growing.

INT: Also the system _____ idea.

BL: I think that's saying the same thing from a different point of view. If you bought a computer system in 1965 you expected it to have a batch processing monitor and a fortrane compiler and a cobalt compiler, that was about it. That was the definition of bay system, very high end bay system actually. If you buy a computer system in 1990 you expect it come with tex windows and a relationable database and a whole bunch of other stuff. That's just standard. You expect to see the level of that continue to rise. That's really where was DEC has done successfully in the past and we'll probably continue to do successfully in the future. The demands for what constitutes a state of the art bay system are constantly growing. So I think that's, you can hang that out in front of all the company's engineering activities as a goal towards which they are ______. You could imagine other things that might, if you conceive of other things as goals for example, you can conceive of setting a goal, trying to make systems better, as easy to use as possible. That doesn't play to company strength. That would not be a sensible thing to set up as a first level goal because we're just not that good at it. Other people are clearly better at it so it would be better just to buy it from them.

INT: I think that captures the strength.

BL: As long as it has to do with history, I'm not sure.

INT: Everything that has to do with the future has to do with the past. This is great. You've shared some wonderful perspectives and observations.

[END OF BUTLER LAMPSON]