Official Transcript of Proceedings

NUCLEAR REGULATORY COMMISSION

Title:	Meeting of the Subcommittee on Digital Instrumentation and Control Systems
Docket Number:	Not provided
Location:	Rockville, Maryland
Date:	Tuesday, June 27, 2006
Work Order No.:	NRC-1109 Pages 1-296

NEAL R. GROSS AND CO., INC.
Court Reporters and Transcribers
1323 Rhode Island Avenue, N.W.
Washington, D.C. 20005
(202) 234-4433

UNITED STATES NUCLEAR REGULATORY COMMISSION

+ + + + +

ADVISORY COMMITTEE ON REACTOR SAFEGUARDS

+ + + + +

MEETING OF THE SUBCOMMITTEE ON DIGITAL

INSTRUMENTATION AND CONTROL SYSTEMS

+ + + + +

TUESDAY,

JUNE 27, 2006

The subcommittee meeting convened at the Nuclear Regulatory Commission, Two White Flint North, Room T-2B3, 11545 Rockville Pike, Rockville, Maryland, at 8:30 a.m., George E. Apostolakis, Chair, presiding.

SUBCOMMITTEE MEMBERS PRESENT:

GEORGE E. APOSTOLAKIS	Chair
MARIO BONACA	ACRS Member
THOMAS S. KRESS	ACRS Member
JOHN H. HICKEL	ACRS Consultant

ACRS STAFF PRESENT:

ERIC A. THORNSBURY

NRR STAFF PRESENT:

STEVEN ARNDT	RES/DFERR
TODD HILSMEIER	RES/DRASP
BILL KEMPER	RES/DFERR/IEEB

ALSO PRESENT:

TUNC ALDEMIR	Ohio State University
TSONG-LUN CHU	Brookhaven
CARL ELKS	UVA
BOB ENZINNA	AREVA
JEFF GAERTNER	EPRI
TONY HARRIS	NEI
ALEX MARION	NEI
GERARDO MARTINEZ-GURIDI	Brookhaven
THUY NGUYEN	EPRI/EDF
JEFF STONE	Constellation Energy
MICHAEL YAU	ASCA, Inc.

A G E N D A

P R O C E E D I N G S

(8:33:29 a.m.)

CHAIR APOSTOLAKIS: The meeting will now come to order. This is a meeting of the Advisory Committee on Reactor Safeguards, Subcommittee on Digital Instrumentation and Control Systems. I am George Apostolakis, Chairman of the Subcommittee. Members in attendance are Mario Bonaca and Tom Kress. Also in attendance is one of our consultants, Dr. John Hickel. The purpose of this meeting is to review the ongoing digital system risk program, and the development of a regulatory guide on risk-informed digital system reviews. The subcommittee will gather information, analyze relevant issues and facts, and formulate proposed positions and actions, as appropriate, for deliberation by the Full Committee. Eric Thornsbury is the Designated Federal Official for this meeting.

The rules for participating in today's meeting have been announced as part of the notice of this meeting previously published in the *Federal Register* on May 25, 2006. A transcript of the meeting is being kept, and will be made available as stated in the *Federal Register* notice. It is requested that speakers first identify themselves and speak with sufficient clarity and volume so that it can be readily heard.

We have received no written comments from members of the public regarding today's meeting. Representatives from industry have requested time to make an oral statement, which we will hear at the end of the meeting. We will now proceed with the meeting, and I call upon Mr. Bill Kemper from the Office of Nuclear Regulatory Research to begin the presentations.

MR. KEMPER: Thank you, George. Good morning. My name is Bill Kemper. I'm the Branch Chief of the Instrumentation and Electrical Engineering Branch in the Office of Research. We're here today to provide an update to the ACRS INC Subcommittee on a research program that will provide modeling methods, tools, data, and regulatory guidance by which the Agency can review and improve risk-informed license applications for digital safety systems in nuclear power plants.

Currently, digital safety systems license applications for digital safety systems are reviewed and approved using deterministic methods in accordance with Chapter 7 of the Standard Review Plan. Now this research program will enable the Agency to also assess the contribution of these systems to plant risk during the licensing process.

Steve Arndt, who works in the INC Group, the Office of Research, will take the lead for today's presentations. He's the Project Manager for this project. Also, Todd Hilsmeier, to my right here, is working with Steve. He's from our PRA Branch in the Office of Research, and he is also managing a part of this project, as well.

They are supported today by staff members from several of our contract organizations. We have folks here from Ohio State University, Tunc Aldemir, and we also have folks here from Brookhaven National Lab, and that would be Louis Chu and Gerardo Martinez. Excuse me. I hope I pronounced that properly. And have I left out anybody else? Is there anybody else here that we want to introduce? Carl Elks is from the University of Virginia, and Michael, who have developed a part of the research program that we're going to use in developing this risk-informed approach here. So we have a lot of material to discuss today, and we really look forward to your insights and feedback on this information.

This research project involves the application of modeling methods for digital safety systems that are relatively new, or at least not used within the nuclear industry at this time, so your advice and counsel would be greatly appreciated during these discussions. I see we have a lot of folks in the room, so there appears to be a lot of interest in this process from others, as well, so look forward to any input that our stakeholders may have, as well. So with that, I'll turn it over to Steve to begin the presentations.

MR. ARNDT: Thank you, Bill. As you can see from the schedule today, we have a number of different presentations, and I'm going to try and get through the introduction quite quickly so we have time for the technical discussions. We're going to go through a lot of different areas. If the members or the Chair would like us to concentrate on certain areas and move more quickly on others, please just let me know, and I'll facilitate that. I'd like to keep the meeting as informal as possible, free exchange of information.

For those members who might need a little refreshing and John, who I don't think has seen this before, I have a few slides just to introduce the research. As Bill mentioned, the research is intended to investigate potential procedures and methods for including reliability models in digital systems in current generation PRAs, develop these methods to the point they can be integrated into agency tools, and developed with necessary regulatory guidance, including understanding what the methods are, and which methods are most usable for this particular purpose, because there are a lot of different digital system modeling methods out there, determine which of these systems need to be modeled in terms of digital systems, how detailed a model, what level of modeling you need to actually put into the PRA, develop and test the methods for realistic applications, and then develop acceptable regulatory guidance associated with that.

CHAIR APOSTOLAKIS: Are you going to address the second sub-bullet today?

MR. ARNDT: We're going to talk about it a little bit.

CHAIR APOSTOLAKIS: Is this what we discussed in the past, the classification of the systems and so on?

MR. ARNDT: It's part of the classification. There are several different crossing classification issues, but one of them is the complexity of the system, and how that dictates both the kinds of modeling methods you need to adequately address them, and what level of integration into the PRA you need. There's other ways of classifying it, depending on other things, but we're going to talk about that a little. That's one the sticking points, and we're challenging parts of this, but we will talk about that at some level. If you have additional questions as it goes forward, please let us know.

Issues facing the NRC - we've been talking about this for a number of years. The licensees are replacing analog systems. The industry has expressed interest in risk-informed methods, similar to those laid out in Reg Guide 1.174 as an alternate method for licensing these systems. However, the research into how to do this does not currently support this application, which is the reason why we have a research program.

In addition, we're starting to run into situations where other risk applications are being limited or could potentially be limited because the general PRA does not model these systems. As we start doing more tech spec updates, et cetera, et cetera, we're having to exclude digital systems from that piece of those applications because we don't have adequate models. And, of course, the agency analysis methods do not at present private any independent means to support that, so we'll talk a little bit about how we're going to, if the research is successful, integrate these in with the current NRC tools.

CHAIR APOSTOLAKIS: Is the industry developing methods along these lines?

MR. ARNDT: Yes. And the industry - I think we talked the last time - has proposed a methodology that we're looking at.

CHAIR APOSTOLAKIS: Oh, yes.

MR. ARNDT: Other industries, like the aviation and space, have proposed methodologies, as well. There are some advantages and disadvantages associated with those.

At our subcommittee meeting in June, the ACRS Subcommittee specifically asked that they be consulted as the program progresses, and that's specifically what the purpose of this meeting is. We have some intermediary products. We've shared some of the drafts with the committee, but this is primarily a progress reporting meeting. We've made some progress, and we want to tell you where we are, get your feedback, get your input on that.

The committee encouraged the review of

software-induced failures, and we're going to hear about that today. The committee encouraged critical review of various methods, and we've published some research in that area looking at various methods and what we consider to be the most effective. And the committee also encouraged the staff to view digital systems from a systems standpoint, while acknowledging that there may be some applications that that's not necessary. And we'll talk about that, as well.

So we're looking at a number of different areas. It's a rather large and complex program, as you might have guessed from Bill's list of people that are working on it. We'll talk a little bit about how all the pieces fit together. We're basically looking at the various methodologies and developing some benchmarks to assess the relative capabilities and limitations of the different methodologies, at the same time informing our development of a regulatory guidance. We'll talk a little bit about the status of the development of the regulatory guidance at the end of the day. That's basically a preliminary issue. We will, of course, bring the draft regulatory guidance to the Committee before issuing it for public comment, so we're at the early stage of that development right now.

CHAIR APOSTOLAKIS: When is this going to happen, Steve, in the fall?

MR. ARNDT: Yes. I have a draft schedule in that presentation.

CHAIR APOSTOLAKIS: Okay.

MR. ARNDT: But one of the things we want to do is get both stakeholder, and ACRS, and industry input into that, so this is your opportunity to give us some general ideas, are we going down the right path. We're also going to probably have a public meeting in August to get stakeholder input to make sure that the conclusions we're reaching are reasonable and appropriate.

CHAIR APOSTOLAKIS: Is this the first time today that you will present this to the public, the regulatory guide?

MR. ARNDT: Yes. It's just first thoughts on the regulatory --

CHAIR APOSTOLAKIS: The ideas, yes.

MR. ARNDT: The ideas.

CHAIR APOSTOLAKIS: But this is the first time.

MR. ARNDT: The first time, yes. Most of you have seen this diagram. John I don't think has. This is just a structural diagram of how all the pieces fit in our program. I'll go through it very quickly.

This first part is basically developing an approach, come up with an idea of how to do it. Supporting that is the review of the failure data, which was encouraged by the Committee, and the review of the current reliability methods, which we talked about in NUREG 69.01.

Supporting the development of the actual analysis is the supporting analysis, understanding how the system works, the failure most effects analysis, the digital system testing, and various other things, and the critical element that a lot of different elements are feeding into, the determination of what systems need to be modeled and at what level. This is an ongoing challenging part.

Now this path is a review and evaluation of dynamic methods. This path is review of traditional methods, fault trees, event trees, and supporting methodologies. The idea here it look at both methodologies critically and understand what systems can be modeled at what level using what methodologies, and what assumptions you have to make, and what limitations you have to make in those analyses. All of those will feed into the regulatory guidance, which we are currently developing, and the development of the supporting tools for the staff.

CHAIR APOSTOLAKIS: Isn't the box on the upper left corner, the failure data, one of the most critical activities here? I mean, why doesn't it fit into both the traditional methods and the dynamic methods?

MR. ARNDT: It does. There's only so many arrows I can put on my chart. It's a critical element for a number of reasons. One, understanding and assessing what data is out there, what the data spread is in issues like that. Also, understanding how you augment available operational history with other information, like testing data and things like that, is a critical part of all of this. It's a critical part of the traditional methods, the dynamic methods, as well as the determination of what modeling methods you --

CHAIR APOSTOLAKIS: But in reading the reports and data and dynamic methods, one gets, at least the way they are now, one gets the impression that these two groups have not communicated, because the data that are -- date, I mean the numbers that are used, or the quantities that are used in the dynamic method report really have nothing to do with the findings of the failure data report. So at which point is there going to be some integration?

MR. ARNDT: Well, I take a little bit of umbrage with nothing to do.

CHAIR APOSTOLAKIS: Epsilon, they have epsilon to do. I mean, if you read the data report, there's all sorts of things that have happened, and this and that, then you go to the dynamic methods and they say now, this is a transition rate, and precision rate, and there is absolutely no reference to what is out there. And I'm wondering -- you know, it's not -- maybe it's something that you intend to do in the future. I don't know. I mean, this is work in progress.

MR. ARNDT: It is work in progress, and we do intend to increase the review of these issues, because it's a critical issue. But I think when we review that piece of it today, you'll see that we are including those issues, the operational history of the system, the available failure information associated with components and other things feed into both the traditional methods and the dynamic methods. We may not be articulating it as well as we could in the report, and we certainly want to continue to have cross-fertilization. But yes, I take your point.

As we have discussed, we're structured for three major outcomes; basically, the determination of what needs to be modeled at what level and what accuracy, the development of an independent modeling capability, and development of acceptable criteria for risk approaches.

So in summary, what we're looking forward to getting from the ACRS is the review of our progress, advice on the best methods, such as what Professor Apostolakis has just given, meaning the discussion we just had, eventual review and endorsement of the proposed methodologies, and eventual review and endorsement of the regulatory guidance. That will be probably this fall or early winter.

CHAIR APOSTOLAKIS: I think, Steve, the middle box there, "Determination of which data systems need to be modeled, at what level of detail", is a critical one, as you know. And you should give it more prominence, in my view. Again, in reading the reports as they are today, one gets the impression again that, for example, the dynamic methods, this it is. We are proposing this, we're going to apply it everywhere. Then you read the Brookhaven report, it's something else.

Maybe there ought to be –- I mean, I understand that this is something that you cannot finish now before you do other things, but maybe if you have a skeleton of it, and everybody refers to that, and everybody understands that this thing is going to evolve as we progress, I think that will go a long way towards pacifying some people, because I mean, admittedly what is in this dynamic thing is fairly complex. And you're scratching your head, saying well, do I have to do this for actuation systems, for example. And there is nowhere there something that says hey, this is for a class of systems that have these problems or these characteristics, and I think that would be –- I mean, I appreciate that it's something that you cannot finalize now, but having some sort of a skeleton - based on what we know, this is the way we're going, and this is where this method applies.

MR. ARNDT: Yes. At the risk of getting ahead of myself, because we're going to talk a little bit about this later in the day, what we're looking at right now, and again, this is preliminary results, we haven't gotten the Reg Guide ready for prime time yet.

CHAIR APOSTOLAKIS: I know. That's why we're here. I mean, I fully agree it's not –-

MR. ARNDT: But the concept is there's going to be a set of characteristics, performance characteristics, if you will, that will lead us to particular modeling requirements that will lead us to - or the industry if they choose to go down this path - modeling capabilities for certain systems, some will have relatively simplistic modeling methodology, some will have an appropriate uncertainty analysis and data requirements, et cetera; some will have a higher level of detail, and some will have a still higher level of detail. That then becomes both a regulatory concern for us, how good does it have to be for which application, and then an economic concern for the industry, what do they want to do? So that's basically the idea.

CHAIR APOSTOLAKIS: No, I know, but all I'm saying is, maybe you can give us some idea where you're going at this point, without waiting to be ready for prime time.

MR. ARNDT: Okay.

CHAIR APOSTOLAKIS: It's okay. I mean, I understand these things, and we all understand that these things are evolving. John, do you want to say something?

MR. HICKEL: Well, I think I tried to ask

Steve this maybe before, but one thing is this –- it's a split between how much resource do you devote to things like trip and actuation systems, versus emergency diesel load sequencers, versus normal process controls?

If I knew what –- do you have a proposed split as to how much attention you're going to put in this area versus that, or is that too preliminary?

MR. ARNDT: Well, there is a couple of different ways to answer that question. In terms of attention from a research standpoint, we know certain things, and we don't know certain things, and we know things at various levels, so we put the most attention to the things we know least about so we can get a level of understanding that's appropriate.

In terms of regulatory side, and I'm not on the regulatory side, but some of my colleagues are here, the issue is, you want to put the most importance on those things that have the biggest potential for risk to the health and safety of the public, because that's our business. So it's a little bit –- I'm not quite sure what you're getting at by the question.

MEMBER KRESS: It looks like a good place for using risk importance measures.

MR. ARNDT: Yes.

MEMBER KRESS: You could do that, even though you don't know the failure rate, you can do a risk importance.

CHAIR APOSTOLAKIS: At the system level.

MR. ARNDT: At the system level, yes.

MEMBER KRESS: Yes.

MR. ARNDT: Both how important the system is, and how complicated it is, and how important it is to get it right, and/or not miss things is part of the criteria associated with what you're going to do.

MR. KEMPER: This is Bill Kemper. If I could just throw something in here. We're going to talk more about during this presentation of a couple of benchmark exercises that we're going to do. We intend to model the digital feedwater system from a current operating nuclear power plant, as well as the reactor protection system, and engineer safety feature system. So we hope by performing a couple of case studies, if you will, and benchmark examples, we'll be able to provide some guidance along the lines of what you're asking here, George.

CHAIR APOSTOLAKIS: Yes. Don't misunderstand my comment. I know that you guys have been thinking about it. It's just that I think you should give it more prominence even now, so the reader will know that we are exploring this area, these kinds of systems, and put it up front in bold face because if you read some of this stuff now and you stop and think what are we trying to do here, you really don't have that help from you. That's all I'm saying.

MR. KEMPER: Good comment.

CHAIR APOSTOLAKIS: Okay. Who's next?

MR. ARNDT: Okay, if you look at your agenda --

CHAIR APOSTOLAKIS: It says Arndt and Aldemir.

MR. ARNDT: Yes. What we're now going to step through is some of the work on the dynamics, a fairly lengthy presentation. Then we're going to talk through some of the data issues, and some of the traditional methodologies in the afternoon, and then the early thoughts on the Reg Guide at the end.

CHAIR APOSTOLAKIS: Okay. So now we have this big package. Right?

MR. ARNDT: Yes.

CHAIR APOSTOLAKIS: Okay. A lot of slides.

MR. ARNDT: Joining me at the table is Professor George Aldemir from Ohio State University.

This presentation is, as you mentioned, a lot of slides. We're going to go through a quick background on why we're looking at dynamic methods, talk a little bit about the first benchmark. As Bill just mentioned, we're going to have a second benchmark. The first benchmark is going to be a system that is more likely to require the dynamic methods. The second benchmark is going to be a system that's less likely to require the dynamic methods. We'll talk a little bit about what it entails. We'll talk a little bit about data, which is obviously a very important issue in this area. We'll talk about the example model that we're going to use to integrate this system, the two methodologies that are being proposed as pilot methodologies for dynamic methods, dynamic flow-graph methodology and Markov; a little bit about if you do this methodology, how you integrate it into a PRA, because the current fleet of PRAs are fault tree/event tree systems, and have an acceptance criteria that's based on Delta CDF or Delta LERF. You need to get those integrated.

We'll talk a little bit about interfacing with the current NRC PRA tool, SAPHIRE; procedures and requirements for reliability modeling. Basically, what we've learned in terms of what's necessary to do this based on how far we've gotten on the benchmark so far, and then conclusions to-date.

You mentioned, I'm trying to sit at the head of this multi-technical research program, so this is going to be focused in on the particular dynamic methodologies, but part of the objective of this is not only to develop the dynamic methodologies, but also to understand where you need them and where you don't need them, and what aspects can be modeled with what kinds of systems, and what the limitations are.

CHAIR APOSTOLAKIS: Since you're talking about an overview, I got a little confused when I read the report, because in Chapter 2, there is a lot of discussion in using the words Markov; for example, 2.4.4 says "Modular Markov chain modeling of the DFWCS." And then much to my surprise, there's a whole Chapter 4 on Markov analysis, so what is the –- I mean, can you give me an overview - in Chapter 2 we are doing this, in Chapter 3 we're doing that, and in Chapter 4 we're doing that. I don't see how what you have in Chapter 2 relates to Chapter 4.

MR. ARNDT: Okay. In that report, and I apologize to the public. This is a draft report that's not publicly available yet. In that report, which is a report that will be published here in a few months, Chapter 2 talks about the system and how we develop data for the system. In that analysis, we use a system model to try to understand what data we need. That system model is a Markov model, so in Chapter 2, we're basically talking about our understanding of how the system works, and based on that, what data we need, and how we generate that data. That's one application of Markov associated with trying to understand the system.

CHAIR APOSTOLAKIS: I mean, since you have a Chapter 3 on the dynamic flow-graph methodology, shouldn't you be using that also to develop whatever data they need?

MR. ARNDT: Yes, but the particular model we're using for understanding the system just happens to be a Markov model. It could have been a dynamic flow-graph model, it could have been --

CHAIR APOSTOLAKIS: So this is not a comparison of the methods then.

MR. ARNDT: No.

CHAIR APOSTOLAKIS: This is focusing on the dynamic model.

MR. ARNDT: The chapter on the Markov --

CHAIR APOSTOLAKIS: Four.

CHAIR APOSTOLAKIS: Three and four are the two different dynamic methods. Chapter 2 is understanding the system and developing the data necessary for the system, how does it fail, what are the failure modes. We just happened to use a Markov model in that analysis of the system. It could have been any state space model we wanted, we just happened to use a Markov model.

CHAIR APOSTOLAKIS: The question is, I mean, if you are producing data for information really about the system in Chapter 2, it should address both methodologies then. I mean, you're already biasing the thing towards the Markov approach. Anyway, is there going to be a presentation on Chapter 2?

MR. ARNDT: Yes.

CHAIR APOSTOLAKIS: Okay.

MR. ARNDT: Okay. As you mentioned, this is a fairly long presentation. Some of it I will try and skim through relatively quickly. Obviously, if there are questions, we can do that, go into detail. Some of it we'll try and talk about a little more detail, but this is basically where we're going.

As we mentioned earlier, we're trying to develop the models to support the NRC policy statement that encourages expanded use of PRA in all areas supported by the state-of-the-art and data. We're developing the various models. We're looking at it from a number of different aspects, but particularly from the system standpoint because that is the preferable way to look at it, and we have been encouraged to do that by this committee, by the National Academy study, and others. However, for the near term, we're going to have to - if we choose to model in a dynamic way, we're going to have to find a way to get back to PRA through some kind of traditional PRA through event tree/fault tree-type applications, so we're also looking at how you get that information into a fault tree/event tree-type of approach. And there's a number of ways out there, we just chose one particular way which we think is particularly encouraging.

We're looking at issues that in this part of the project, the dynamic part, that might drive us toward using dynamic methods. Particularly, dynamic interactions between the system and the process that it's involved with in case of a controller, in particular, the physical processes associated with it, as well as internal issues within the digital systems that are either sequential or time-based, or things like that. These we refer to, for convenience, as Type 1 and Type 2 interactions. Some systems, as we mentioned earlier, will have relatively few Type 1 interactions. Actuation systems that just meet a threshold and do a particular action, don't have a lot of process feedback in them. Control systems have a lot of process feedback in them. Depending upon the complexity of the digital system, they may or may not have a lot of Type 2-type interactions. If there's a lot of communications between the different internal systems, if there's data sharing, if there's multi-tasking, there's a potential that there's going to be a lot of interactions that will be sequence-dependent, or time-dependent, and will need a more complicated model.

For example, the Turkey Point generator sequencer failure that occurred several years ago, where the system was in diagnostics, and got a real actuation signal, and failed to drop out. That is an internal Type 2 sequential issue that you need to address in some way for that kind of system, if you're going to have a lot of diagnostics, or if you're going to have a lot of fault checking, or if you have a sequential logic that could have timing-dependent failure modes.

CHAIR APOSTOLAKIS: To what extent are these systems being used now in safety systems?

MR. ARNDT: It depends on the plant, depends on the particular safety system. There's not been a - let me see if I can say this correctly - there's not been a RPS or ESFAS update in a digital system under the new regulations.

CHAIR APOSTOLAKIS: There has or has not?

MR. ARNDT: Has not.

CHAIR APOSTOLAKIS: Has not.

MR. ARNDT: There has been some safety systems that have been upgraded with digital systems, but they're not RPS or ESFAS.

CHAIR APOSTOLAKIS: And these are just actuation systems, or there is feedback there, and control?

MR. ARNDT: There are feedback systems, simple control systems.

CHAIR APOSTOLAKIS: And the staff has approved those? I guess they have.

MR. ARNDT: Using the deterministic rules.

MR. HICKEL: Hey, George, CE has been running digital protection systems based on stored computer software since 1978.

MR. KEMPER: Yes. This is Bill Kemper, again. Yes, there are many digital applications out there. The CPC Plant Protection System that he just mentioned, for example, is one that's been around for a long time. There's currently digital devices being put in place to replace other antiquated digital systems under 50.59. Very few have been submitted to the agency, though, for license amendment approval, if you will. However, as you're well aware, the Oconee application is really the first full-blown RPS and ESFAS upgrade from analog to digital technology, so that's what we're really dealing with at this point. But as an example, for example, at Palo Verde, they replaced their platform with an ADVENT 160, the "Common Q" processor. Oconee has got, I understanding, in their QB system, TELEPERM, so there are examples of equipment installed out there, but it's not on a very large scale yet. We're just kind of at the beginning of that bow wave, if you will.

MR. ARNDT: And there's a significantly larger fraction in the non-safety side.

CHAIR APOSTOLAKIS: Yes.

MR. ARNDT: Okay. Again, I'll briefly talk about this. This is basically the chart I showed before. This side is the dynamic part, which is what we're going to talk about today, but it also has interactions with these other supporting analysis; particularly, of course, the determination of what systems need to be modeled.

So the objective is to develop procedures and methods for incorporating these reliability methods into a PRA, and what we're doing is we're doing pilot studies, as Bill mentioned, to understand if the proposed methods are capable of modeling the systems adequately, and what are the limitations associated with it. And then understand how you integrate those into the current regulatory structure for risk-informing systems that the NRC has, the 174, Delta CDF and Delta LERF issues for INC, and also look at other deterministic rules associated with that.

So this is basically just words associated with what was in that bubble chart; investigate the applicability of current methodologies, review the limitations and advantages of dynamic methodologies, review what other people have been doing, the railroads, space, industry, NASA and other things, review the existing regulatory framework, identify the minimum set of requirements, or at least a preliminary minimum set of requirements, which is going to get evolved as we learn more about how these systems work; take those methodologies, see whether or not they meet the requirements that we've identified, and then test them with benchmarks, so we've done a preliminary review of the first six of those steps, and determined that the two leading candidates from a dynamic standpoint are a Markov methodology, and a dynamic flow-graph methodology. Each has limitations and advantages both in terms of modeling complexity, the data you need, how you structure it, the amount of information that's necessary, the amount of quantitative versus qualitative information you get. And we're getting leaders in both those areas as subcontractors and contractors to look at that methodology.

Okay. The next three or four slides are just a review of the benchmark we chose. The purpose of this is to talk about why we chose this particular benchmark, and how we've set it up. The idea is to have a benchmark that hits the various possible modeling requirements as much as reasonable for a single system, because we're not going to do 30 systems to make our decision. We want to do two or three systems to make a reasonable assessment of what's really necessary for practical systems, so we chose the benchmarks in such a way that they're both representative of real systems, and they have a lot of the characteristics of various digital systems, and the feedback processes associated with them.

This particular benchmark is a digital feedwater control system based on an operating plant's digital feedwater system.

MR. HICKEL: Which plant?

MR. ARNDT: I'd rather not say in a public meeting.

MR. HICKEL: It's a real one, though.

MR. ARNDT: Yes. We've taken the actual system, we've generalized it a little bit to be representative of this type of system; that is to say, an important to safety, but not safety system that has interactions with the process, and interactions within itself between its component parts. Basic purpose of the feedwater control system is to maintain the level in the steam generators.

For the particular scenario we chose, the failure criteria for this particular system is above 30 or below 24 inches. This is scenario-dependent. We'll talk about the particular scenario we chose later in the presentation.

MEMBER KRESS: Was there a reason for those numbers, like the steam generator loses its effectiveness beyond that or something?

MR. ARNDT: Based on the particular scenario, there's numbers –- some other actuation happens, it either loses its effectiveness, or causes another system to actuate or whatever. Connected basically to the main feedwater system that regulates the feedwater pump, the main feedwater valve and bypass valve. The controller in the system's basic purpose is to regulate the steam generator, level the temperature, and deal with other things associated with the steam system.

Real quick overview - steam generator system, obviously, there's booster pumps and condensate pumps in here, but just simplified system. You have inputs, power from the reactor, steam flow level, feed flow, feed temperature. The system is basically structured with a main computer and a backup computer, a controller which takes information from these computers for the bypass valve, the flow valve, and the feed pump. You have the back-up controller, and I'll talk a little bit about how that's configured.

You have a number of different internal inter-connections. This is the Type 2 interactions that I mentioned. The main computer will trip off to the back-up computer. It also has a watchdog associated with the various controllers it is providing information for. We also have something known as a --

CHAIR APOSTOLAKIS: You know, when you use terms that are not commonly used by everybody, you should explain that. Watchdog status - I mean, what does that mean? It's probably part of the language of this field.

MR. ARNDT: Apologies. Watchdog timer or watchdog status is a commonly used fault tolerant capability among digital systems. The concern is that you either get stuck in the loop, or if you hang the computer, or you do not progress through the system, watchdog, you can configure it in a number of different ways, but in this most basic configuration is waiting for certain things to happen. If it doesn't happen under a certain time cycle, or under a certain set of conditions, it will flag an error, or trip the system out, or go from a primary system to a backup system.

CHAIR APOSTOLAKIS: Good.

MR. ARNDT: The only point of this slide is basically there's a number of different internal connections associated with how the system works, how it feeds from one system to another, what the fault tolerant capabilities are, if the main computer does not continue to update, the controllers will take the last signal. It will identify issues to the operator that will allow the operator to go into manual mode, between the different controllers going between the various modes of operation, full power and low power operation. The point being, there is indications associated with it that lead us to have Type 2 interactions in the system.

The input parameters are cross-tied based on the various channels, as you would expect, to reduce the likelihood of single failure criteria. Control laws are non-trivial, and I won't go through all these in detail, but they have a number of fairly complex control laws associated with the demand, the compensated air, and the level, both for the flow, the level, the power, the positions for all the different valves, and the speeds. The point here is, there's a lot of process dynamics that can feed back into the control system that makes when the system fails and when which pieces of the system fail important.

CHAIR APOSTOLAKIS: So these laws are used by either dynamic methodology?

MR. ARNDT: These laws are used by the dynamic.

CHAIR APOSTOLAKIS: Are they also being used by DFM?

MR. ARNDT: They're being used by both of the dynamic methodologies. This is the system. We'll talk about how we model the system in both the dynamic methodologies later in the presentation.

MR. HICKEL: I guess one question is, is this system taking the original PID controller and converting it to an equivalent digital, or is this something that's a revolutionary system that's trying to feed forward, or something like that?

MR. ARNDT: It's basically a conversation of the PID controller that was originally in there. There's some added features, but basically that's where we are.

This is just some more basic information on the control laws. The issue here is because of the way the control laws are developed, the current state of the system is dependent on the historical information in the digital system, so there's history in the state space.

As I mentioned before, there's a number of fault tolerant capabilities in the system. One of the reasons we care about this is, it touches on a lot of the potential reasons why you would need a dynamic methodology, the DFM, the Markov, or something else, as opposed to a simple fault tree/event tree. So the controllers for the main feed valve, backup feed valve and the feed pump for the control systems to the corresponding feed control points provides fault tolerance in case the computers fail, gives the operator time to intervene, switch from automatic to manual. The computers are independently wired to different power sources. You can have different kinds of single failure controllers, single failure modes. The algorithms take a relatively short time compared to the response frequency, the physical process. There's a watchdog timer, as I explained earlier, on each of the two computers, the backup and the main computer. If the set point – if the system fails, the computers will fall back to a pre-programmed set point value. Each of the computers has a validation and verification of the inputs, so that there's a number of different fault tolerant features associated with the controllers that may lead to Type 2 dynamic interactions.

CHAIR APOSTOLAKIS: So these are included in the two methodologies? They said yes.

MR. ARNDT: I'm sorry. Again, the input ranges are checked, the backup computer propagates the sensor data.

MR. HICKEL: What's a PDI controller? I

know what a PID controller is. Is that just -- is that a typo on the --

MR. ARNDT: No, that's really what it's called. It's --

MR. HICKEL: Portional Derivative Plus Integral, instead of --

MR. ARNDT: No.

MR. KEMPER: No, this is Bill Kemper. The particular plant where this system is deployed, that controller normally monitors, if my memory serves me right, differential pressure across the main feedwater valve, so it's called PDI. It's an indicator. In the fail mode, it reverts to a control device for one of the SD's head, either the main feed valve or the bypass valve controller.

MR. ARNDT: It serves for the purposes of the dynamic interactions, as basically a backup to the other controllers in the system.

As I mentioned as we were going along, the system incorporates all the properties of a loosely coupled system; that is to say, it has a lot of the properties we care about when we're trying to determine what level of modeling detail we need to address. Some of the properties it doesn't incorporate, but those systems may not be important to the kinds of controllers and digital systems that are actually in nuclear power plants. When we wrote the issues for digital systems, we wrote them as general as possible, so we included things like networking and shared external resources.

Without knowing what the licensee is going to bring to us in terms of a configuration, we wanted to be as general as possible. We understand that most, particularly safety system, digital systems are going to be used in a real-time safety system. We're not going to have networking resources, or shared external resources, so that may be a less important criteria which will eventually drop out of a regulatory guidance. We wanted to start general, and focus in.

MEMBER BONACA: I have a simple question here, Steve.

MR. ARNDT: Yes, sir.

MEMBER BONACA: You know, some plants already have this system, this feature. Has any plant attempted to model in their PRA these control systems?

MR. ARNDT: There are models of control and protection systems in PRAs. They tend to be, and I don't know what all 103 PRAs look like in detail, some of them are very, very general.

MR. KEMPER: Black box.

MR. ARNDT: Black box, and most of them, I would say, are incorporated as sub-components of the system as a whole. There are some models, some of them - I'll use a non-U.S. example to be safe, such as the Seiswell B model, is fairly detailed. Seiswell has a fairly detailed PRA model of their control and instrumentation systems, and protection systems. They're not a dynamic model, they can't capture the kind of dynamic interactions we're talking about. Do they need to? Well, that's part of the reason we're doing the research, is to see whether they need to or not. But most of them are fairly general, and some of them are very black box, as John mentioned.

MEMBER BONACA: Yes. Okay, thank you.

MR. ARNDT: As I mentioned earlier, the system includes system history as part of the control laws, so there are opportunities to create artifacts and/or create situations where the exact timing and sequence of events might be very important.

At this point, I'd like Professor Aldemir, who did this particular analysis, to walk you through an example of what can happen in this case associated with timing failure sequences.

MR. ALDEMIR: In the first slide here on

the left, you're seeing the normal behavior of the system. Incidentally, this is simulating a situation where the initiating event is a turbine trip with main computer failed. And the reason why it's failed, is so that the state space is limited for illustrations. This example is taken from the report that we just went through earlier, and in this report, we are trying to illustrate how these methodologies work, and for the ease of understanding, we chose a simpler system with a smaller state space, so it does not represent the whole controller. That's why we purposefully assumed that the main computer failed, to reduce the state space.

So here you see the normal behavior of the system, level starts -- okay. The scenario is such that we're operating at full power, turbine trips, and within 10 seconds the power is reduced to 6.6 percent of nominal power with feedwater flow following, so you have these oscillations until the level stabilizes around 100 seconds. Incidentally, these time constants may not really refer to the actual plant, but these are time constants still lead to believable behavior of the system, credible behavior of the system.

MR. HICKEL: Could I ask a question?

MR. ALDEMIR: Sure.

MR. HICKEL: You say it's a turbine trip.

MR. ALDEMIR: Yes.

MR. HICKEL: Are we talking a plant that has a big steam bypass system?

MR. ALDEMIR: Not to my knowledge.

MR. HICKEL: I don't understand the level - to understand the level in the generator, you've got to know what the pressure is doing, so if you trip the turbine, you've taken away the load.

MR. ALDEMIR: Right.

MR. HICKEL: Steam wants to go somewhere.

MR. ALDEMIR: Right.

MR. HICKEL: If you don't take it somewhere, pressure is going to go way up, level is going to go way down. How is that just oscillating --

MR. ALDEMIR: We are tripping -- the reactor trips.

MR. HICKEL: Right. Okay.

MR. ALDEMIR: So within 10 seconds or so, the power is down to 6 percent. That's where this scenario starts. So at the beginning of the scenario, at least as I've shown on this slide, power is 6.6 percent of nominal, which is 1500 megawatts, and then feedwater is at that nominal flow. Then through the bypass flow valve, in this situation, the main flow valve is not active. The bypass valve is active. It is trying to regulate the flow so that it reaches the set point. I mean, it stays at the set point, which is by convention, zero.

CHAIR APOSTOLAKIS: These are the results of the solution to what, to the laws that you showed us earlier?

MR. ALDEMIR: Not all equations -- this particular initiating event, according to the control laws, is such that only three or four of those equations are relevant.

CHAIR APOSTOLAKIS: But this is the output of what?

MR. ALDEMIR: Part of the equations that you saw in the earlier slide.

CHAIR APOSTOLAKIS: And anything else?

MR. ALDEMIR: I'm not sure if I'm following.

CHAIR APOSTOLAKIS: You're talking about the steam generators --

MR. ALDEMIR: Oh, oh, oh, I'm sorry. Yes. Well, thank you for the remark. In those equations then, I don't know how easy it's going to be for me to go back in slides, but in the equation that governs the level change, there is feedwater flow input, and steam flow out. And these are, of course, related. Now this relation is described by a steam generator module, which was developed -- the one that we're going to use is developed by our subcontractor, ASCA. Also, the developers of the dynamic flow-graph methodology.

In this particular example, we are not using that steam package because, as I said, for simplicity of illustration or the ease of illustration, we are trying to put down equations that you can easily follow, so in this equation, the steam flow is assumed to be constant, and the feed flow is used through a simplified pipe and valve model, also taken from NUREG 64.65, which illustrates how the dynamic flow-graph methodology works. Thank you, Professor Apostolakis. I missed that process part.

Now here, this is very interesting, and actually, it was a surprise for us, too. If you notice, up to 600 seconds nothing happens here. Everything is beautiful, everything is maintained at zero level. If you let it run longer, suddenly you have a kink in the system, suddenly through this control. Now this was by accident. Turns out that our colleague who was doing the programming put an artificial or unnecessary bound on one of the parameters, and it's basically an artifact. The real system does not do that, if you program it carefully. But well, we are trying to model software faults, so this is the kind of experience that you can have with the model. Incidentally --

CHAIR APOSTOLAKIS: Your own people make mistakes?

MR. KEMPER: Hard to believe, isn't it?

MR. ALDEMIR: Well, I mean, it was fortunate, because then we created an artifact in the system without intending. Incidentally, these types of events have been observed in real life. And in the report that was being referred to earlier, we cited about four or five examples, where these kinds of events were observed in plants either through the process, complexity of the process, longevity of the process, or actual error in the tuning of the controller. So the benchmark does capture these type of events. Well, I'll come to that later on, but talking about the requirements - can it produce observed failures? Yes, this is one of the cases where we can produce observed failures, because these things have been observed in actual plants.

Now another interesting thing here is that - and this is, again, not intentional. We did not choose the parameter so that we'll have this behavior. It just so happened that we did have this behavior, the discoveries were accidental, too.

In this situation, bypass flow valve, we took curves here. Let's take the first one. The blue one, the steam generator is chugging along, and the level is changing. And at 43 seconds, bypass flow valve fails stuck, and you have a low level. If the bypass flow valve fails stuck at 44 seconds, you have high level. One second difference, two different failure modes.

MR. HICKEL: The valve was modulating, obviously?

MR. ALDEMIR: That's exactly right. And the stuck mode is such that it just gets stuck and so it has to refer back to the history-dependent information, and just so happens at that time, exactly where the level is, you may have totally different modes.

CHAIR APOSTOLAKIS: So what do we learn from this?

MR. ALDEMIR: We learn from this that it is very important to model the timing of events in the reliability model, so it's an illustration of why we may need dynamic models.

CHAIR APOSTOLAKIS: A one second difference?

MR. ALDEMIR: One second difference. And as I said, this wasn't intentional. Purely by accident, we chose the time clusters for the system. We did an analysis. I don't think we have it in the slides, but it is in the report. We did a little analysis of the controller to see what kind of parameter ranges will lead to stable behavior, and arbitrarily chose time constants, and just so happens that this is the type of behavior we observed.

CHAIR APOSTOLAKIS: What do you mean by "time constants"? Which one did you choose --

MR. ALDEMIR: If you go to the -- again, I don't know how easy for me to switch, but if you look at the original equations, there are a number of controller parameters.

CHAIR APOSTOLAKIS: Okay. Okay.

MEMBER KRESS: Couldn't you consider either one of those paths a failure, and not have to know that time --

MR. ALDEMIR: Yes, we may have to. For example, I mean, in this situation, I hope I'm recalling this correctly.

MEMBER KRESS: I'm very skeptical about one second timing.

MR. ALDEMIR: As I said, it was surprising to us, too. But that's what we have observed. Incidentally, this type of difference in failure modes is not the first time that we're observing in this system. We have a publication in 1989 where we are using the HIPCO system, bleed cooling of BWR. This is NUREG 69.01, where again, the timing of events are very important, and it can take you to high level or low level.

MEMBER KRESS: Would you do something different depending on which of those modes --

MR. ALDEMIR: Yes. For example, in this situation what happens is that if we hit the low level - now I hope I can recall this correctly - if we hit the -- right now we are dealing with the bypass flow valve, turbine is not available. So if we hit the low level -- sorry, we are dealing with the auxiliary system, I think. We hit the low level, and then the turbine is made available as a heat sink, and then the main flow controller comes into play. And if we hit the high level, I'm assuming that this is going to be the performance of the steam generators. So in the

HIPCO system that we used earlier, if you hit the low level - now that becomes a safety-related action, because it actuates the LPCI system or LPCS system. So if you hit the high level, you don't do anything.

MEMBER KRESS: Explain to me why the high level is a problem.

MR. ALDEMIR: High level, I presume, this is the steam dryers performance deteriorating.

MR. KEMPER: This is Bill Kemper. Yes, this plant is a PWR with U-tube steam generators, so high level, the problem is just as Tunc said, the dryers and everything becomes immersed in water, carry-over and damage the equipment.

MR. ALDEMIR: So the failure mode is important in the sense that, in general, because one may lead to a safety-related action.

CHAIR APOSTOLAKIS: But, I guess, I'm thinking, again, in terms of traditional modeling. The two failure modes would be recognized by the analysts, I think, if they lead to different sequences. And, again, is the issue of the timing, 43 versus 44 seconds, important, as long as they recognize that different things may happen, depending on whether you're high or low.

MR. ALDEMIR: If we are running a

qualitative analysis, you are right. Now if we are doing a PRA and quantifying it, it makes a lot of difference in quantification whether you go to one failure mode or the other failure mode. And we have --

CHAIR APOSTOLAKIS: Yes, but the guy who does an event tree will do that. He will just -- the only thing he will ignore, the way I understand it, is the fact that there is a difference of one second there to go to one to the other, but you will have this mode and that mode.

MR. HICKEL: This is not unique to digital. I could postulate the same kind of issue on an old analog system. The feed reg valve - if the reg valve locks up, it's going to either fail high or fail low. The relevance to digital is what I'm trying to understand.

CHAIR APOSTOLAKIS: Yes. But isn't it correct, though, that if you do a PRA and you recognize that there are two failure modes, you will have them there. What you will not have is the timing, and if the timing is important, I bet you a good PRA analyst will find a way to include that there, too. Now just one second difference --

MEMBER KRESS: I could see where the

timing, though, my affect the liability probabilities.

MR. ALDEMIR: That's right.

MR. ARNDT: There's two primary issues, yes. In all likelihood, if you've done a good failure modes and effects analysis, and know the different kinds of failure modes you might end up with, in a traditional fault tree-type analysis, you'll have these different failures. There's two issues. One, depending upon the complexity, this is actually a relatively simple set of scenarios. There are some scenarios that are much more difficult to see just by looking at and trying to analyze and see whether or not you have captured all the different failure modes. Simple systems, much higher probability you're going to capture all the failure modes; more complicated systems, more interactions, more dynamics, less probability.

The other thing is, as we've talked about, if you're trying to quantify the system, it's much more difficult to get a good quantification if you're not including all the characteristics of the system, such as these characteristics. The point is, we're trying to understand what factors may influence the level of modeling detail that's necessary. Okay?

To answer John's question, a lot of these things - well, actually, the vast majority of these things are associated with system complexity, not necessarily digital, although there are some things that are digital-specific. The fact is, because digital systems tend to be more complex, at least at the micro level, you tend to run into more of these issues. It doesn't mean you can't make a very simple digital system. Okay?

PARTICIPANT: Deja vue, wonderful timing, one of George's big issues.

MEMBER KRESS: We'll let Mario be --

MR. ARNDT: I'm going to go through three or four slides here. This was the issue that Professor Apostolakis brought up earlier associated with how we are structuring understanding the system in terms of what the data is. And in any basic data generation or data gathering process, you want to have a systematic methodology to look at what data you need, which is dependent upon both the system and the model you're trying to generate the data for. You choose the model of the system that is reasonable for the level of detail you need. You choose plausible modeling assumptions associated with that. You look at all the parameters that need to be modeled in a logical way and you work through the process, understanding the uncertainties, and trying to understand the critical parameters, and the statistical information necessary to get a good confidence bound on that system.

Like any system - in this case we happen to be choosing two dynamic methodologies, DFM and Markov - you need models that are supported by observable credible data. In this particular case, what we start with is historical plant data and database information for the components. In this case, we looked at the RAC Prism database, there are other databases out there. You then go and look at the specific plant data, if you have any. This is important, particularly in digital systems, because you have to map the entire input space. And in George's parlance, the context of the system. In traditional digital or software modeling, you usually talk about the operational profile. It's basically the same concept. What is the space of all possible inputs, and what's the probabilities associated with those?

You can get a lot of that information from the plant historical data, if you happen to have it. The information you don't have, or need additional information on it, you look at other mechanisms associated with it. In terms of hardware, you might look at stress testing of the system or environmental testing of the system, in terms of digital systems you usually look at different kinds of stress testing of the system, or testing of the various possible failure modes associated with it. The methodology we chose, which we happen to like, but is not the only way to do it, is a fault injection campaign, which looks at the potential failure modes, both safe failure modes and unsafe failure modes, and then maps back through a system model, in this case the Markov model, the potential input spaces that are necessary to get those critical output failures. But the purpose here is simply to augment the data, get a good understanding of what the failure rates likely will be.

CHAIR APOSTOLAKIS: Now there is a number of diagrams and discussion in the report that I don't see you having here, so when would be a good time to raise the questions?

MR. ARNDT: Give me two or three slides. If you have additional questions, we can do it there.

CHAIR APOSTOLAKIS: Okay.

MR. ARNDT: If you'll note, at the very end of that package, we have additional backup slides to talk to these issues, if you want to talk to them.

MEMBER KRESS: On this slide, though, I presume coverage means that part of the input space you didn't fault inject or what? Could you explain what "coverage" is to me? Let's put it that way.

MR. ARNDT: Coverage is a generic term used in digital system modeling analysis. There's several different ways you can model it, but it's basically a determination of the likelihood that you're not going to detect a failure mode based on the test that you conducted.

MEMBER KRESS: Because you can't do all the range of inputs that are possible.

MR. ARNDT: That's correct.

CHAIR APOSTOLAKIS: This is where I have a problem with the report. On page 2-30, there's an incredible statement. "Suppose if we test and get no undetected failure modes, by the fundamental law testing, testing reveals the presence of errors, not the absence of them. We must establish a lower bound for the non-coverage one minus C termed with a non-zero number. What is often done is to assume that one undetected failure occurred in the testing." This is incredible that we see something like this now. We've been discussing this in PRA space for decades, and to say that I have zero failures; therefore, I will assume one is just something -- and then it says, "This assumption has a well-founded statistical theory and legacy, Reference 54", which I found. And the title reference is "Estimating the probability of failure when testing reveals no failures", and I couldn't find anywhere the suggestion that you assume one failure. So this is a completely false statement, and I don't know why it's being made. And as far as I'm concerned, it undermines the credibility of the whole thing.

MR. ELKS: If I may --

CHAIR APOSTOLAKIS: Yes, you may. You can come to the microphone, identify yourself.

MR. ELKS: Carl Elks, University of Virginia. We put that section in there, and I'll be the person identifying myself as citing that reference and using that. That was Dr. Dave Nichols at the University -- I mean, at William and Mary University, who I was working with at the time when we were doing this type of work.

Essentially, this is a software testing technique that has tried to establish through Bayesian methods when you are trying to test something and you do not get any type of estimation of any type of failures, what's the worst case that you can do on this? Now this was applied on a number of different software testing techniques, as well, on fault tolerance techniques. That's why I stated the case that there is a legacy of using this. We have used this, also, at the University of Virginia on several different fault tolerant architectures when we did lots and lots of testing on them, and we found no errors to establish, again, a bound for this type of thing.

Now does that mean that we're going to use that particular technique all the time? No, that was a suggestion that we could use based upon this type of model that we're working on, so I'm not suggesting to the committee at all that this particular technique is the only technique we can use. I'm suggesting that that has been used. It has some statistical reference in legacy in the assessment of safety critical and reliability systems.

CHAIR APOSTOLAKIS: But the paper that is being cited is a rigorous paper using Bayesian methods deriving distributions using zero failures or findings. And if one wanted to be conservative, one could select a percentile of this distribution and use that, and not assume that there is one failure, which is something that really is arbitrary as anything. So

I -- anyway, okay.

MR. ELKS: Okay.

CHAIR APOSTOLAKIS: Thank you.

MR. KEMPER: This is Bill Kemper, very good comment. Thank you for the comment, George. Thank you.

CHAIR APOSTOLAKIS: There are many other questions I have on this particular section, 2.4.2. And I don't know what the best way is. Again, and I have asked this question in the past - there are three states. Okay? Normal, fail safe failure, dangerous. And then it says, "Associated with each state transition is a parameter that indicates the rate lambda at which the failure occurs. And again, I'm trying to understand, what does that mean? And then an hour later, I read the BNL report on data, and they say that they found a 36 percent of failures due to requirements analysis, 27 percent are due to faults that are introduced during upgrades or modifications. And I'm scratching my head now, does this lambda include these things? What does it include, is it hardware failures only? I mean, on the one hand, I have BNL telling me that 36 percent of failures are due to requirements, which I knew, maybe not the 36 percent, but I knew it was a pretty high percentage.

And now I see a transition rate that tells per unit time, there is a constant probability of going from this state to that state. And we have raised this issue before, that before we jump into these Markov models, we really have to scrutinize the meaning of these transition rates. I mean, it's a convenient mathematical tool, I admit, but what does it mean?

MR. ARNDT: Okay. Let me try and address this briefly. Obviously, if you want to go into a lot of detail, depending upon the amount of time we have today, we can have a separate discussion on this specific issue, if you like. But the work that's done by BNL is looking at specific - how you add up those different failures, what kind of failures are they, what kind of failures you need to look at, et cetera. The Markov and DFM modeling methodologies are system-based modeling methodologies. They look at how does the system as a whole fail, so the various failure rates, and we don't need to have them be constant failure rates, they can be - or transition rates. They can be non-constant, if we choose to. We simply are using that as a methodology right now, but if the data indicates that we need time-dependent failure rates, we can do that.

Looking at how you transition from one

state space to another, those failure rates, or transition rates, depending on whether it's going to a fail state or not, are a particular failure. The stuff we're talking about in the BNL can be caused by a number of different things. It could be caused by hardware failure, could be caused by a system failure, could be caused by interaction between the hardware and the software. What we're trying to do in the BNL failure database work is understand how do you populate that failure database, and what has to be included in it? Some of those will be failures that are driving a system from one state to another.

CHAIR APOSTOLAKIS: But, Steve, if we have design errors where design is used in the broader sense, includes requirements, includes specification errors and so on, and these are a significant percentage of the observed failures in the past, failure rates do not account for those, because with a failure rate you are saying my system is working now, and there is a certain probability per unit time that it will move to some other state.

MR. ARNDT: Correct.

CHAIR APOSTOLAKIS: Here it's working now, but if it enters another regime where there is, indeed, a specification error, it will not work, period. There is no -- so what is the time? Is it the transition rate to that regime, in which case the fault manifests itself?

MR. ARNDT: Yes, exactly.

CHAIR APOSTOLAKIS: But that's the kind of thing I'd like to see in these reports. I mean, don't just throw out this is -- then there is other statement, "The probability of being in a fail safe state or a fail unsafe state can be solved using sarcastic Markov modeling." How on earth do you know? What do you mean, that's a postulate on your part. This scrutiny of the assumptions is something that I would really like to see, and have a detailed scenario of what we mean by these failure rates. And when you have -- if you look at the BNL report, for example, and you say yes, this is the rate of going into that area where there may be an error, pick a few and see whether that kind of interpretation or explanation makes sense, because we are really -- I mean, this is very important stuff, and there is a danger here, not that you guys are doing that, of course. I don't expect you to do that, but it's the danger that because there is a model some place, we're going to force this -- you know the Procrustian bed?

MR. ARNDT: Yes.

CHAIR APOSTOLAKIS: Okay. Everybody knows about the Procrustian bed now. So that's good, so this is the kind of thing that bothers me when I read this.

MR. ARNDT: Okay. We can articulate that much better.

CHAIR APOSTOLAKIS: I mean, the CIs, and the other question, of course, is okay, I inject the fault, I find the problem. Don't I fix that if I find the problem?

MR. ARNDT: Yes, you do.

CHAIR APOSTOLAKIS: So how does that play into all this? I mean, if every time I find an error - you see, in standard PRA with hardware failures - okay, the pump fails. We expect that, it's random failures and so on. The nature of the problems you are finding here is different.

MR. ARNDT: That's correct. It's different.

CHAIR APOSTOLAKIS: And you'd fix them, so the question is now what do I do after I fix them? Do I say I found three faults, but then I fixed them, so what's going on here? By the way, NASA has the same problem as we speak, because they fix everything. Okay? They change the design of the system, and some people claim then the past record doesn't apply.

MR. ARNDT: And there's really two things we're trying to understand to support these kinds of modeling issues. One is, what is the likelihood of faults remaining in the system we've tested, and there are methods associated with that. And the other thing is, what is the likelihood that we haven't tested everything, which is basically the coverage concept. You develop a structure by which you go from the failed states that you know would be bad, through a model to understand what input space you need to test, and you test a significant fraction of that.

CHAIR APOSTOLAKIS: No, I understand that, and I think it's a very difficult problem. I mean, the step of measuring, go to a model, and what kind of model. But I'm not saying that the fault injection method is no good, but you really have to be careful what information you're getting out of it, and how you're going to use it.

MR. ARNDT: Exactly.

CHAIR APOSTOLAKIS: Not arbitrarily say I'm going to assume this, I'm going to assume that, and keep going. I mean, that's not - especially in this regulatory space, that's not the way to do things.

MR. ARNDT: Right. As I think I mentioned earlier, the tool that we developed, obviously, for our independent assessment may not be the same tool or same strategy that the licensees choose, but we want to understand the capabilities of the various methodologies.

CHAIR APOSTOLAKIS: Now there is a table of failure rates presumably produced by default injection method on page 2-34, and there are some -- I mean, the rates are on the order of 10 to the minus 6 per hour, but two questions here. One, they seem to be focused on hardware components. They don't include software failures. Right? Is that correct?

MR. ARNDT: This particular methodology looks at the system as a whole.

CHAIR APOSTOLAKIS: But these components are part of the controller. Right?

MR. ARNDT: Yes.

CHAIR APOSTOLAKIS: But it does not -- they don't include software faults, where all the components are working but there is an error --

MR. HICKEL: You've got a bug.

CHAIR APOSTOLAKIS: Yes, you've got a bug.

MR. ARNDT: Right. That particular chart does not, no.

CHAIR APOSTOLAKIS: It does not.

MR. ARNDT: But the methodology looks at any kind of failure, and then it traces it backwards through the system to determine whether or not that failure manifests itself by a software bug, a firmware bug, a hardware bug, a random failure of whatever. This particular one did not do that.

CHAIR APOSTOLAKIS: Okay. Now again, when you see something like that, there is a great temptation to go to the BNL reports. And on page 14 of the collection of failure data, there are all sorts of failure rates for various components, and how do they compare with this table, 2.4.1 in this report? This is the kind of coordination, it seems to me, that maybe you haven't done yet because these things are still being produced, but at some point, you can't have a table in the report from BNL that has numbers for all kinds of things, and then another table with different numbers, unless there is a reason.

MR. ARNDT: Yes.

CHAIR APOSTOLAKIS: If there is a reason, then that's fine. So that's a comment here, that these reports, they have to feed into each other.

MR. ARNDT: Yes. Absolutely.

CHAIR APOSTOLAKIS: And the BNL report, of course, reports actual events.

MR. ARNDT: Yes.

CHAIR APOSTOLAKIS: As opposed to producing using fault injection methods and so on, which on the other hand, is very system-specific, which has a great value.

MR. ARNDT: Yes. Exactly.

MR. KEMPER: This is Bill Kemper. If I can just interject something here; we do intend to go through the BNL information in much more detail, George.

CHAIR APOSTOLAKIS: Good.

MR. KEMPER: So maybe some of these questions might be answered as Todd and BNL goes through that information.

MR. ARNDT: Okay.

CHAIR APOSTOLAKIS: But again, Steve, in Chapter 2 of this report, using whatever method, there are failure rates of components and coverage factors, and all these refer to hardware. Is that correct? No faults in logic, or bugs, or whatever.

MR. ARNDT: The point of this report is to demonstrate the methodology, not to talk about the results. There will be a subsequent report that talks about the results of this benchmark.

CHAIR APOSTOLAKIS: No, I understand that, but if the methodology is limited to hardware failure, that's something we want to know.

MR. ARNDT: No, it's not.

CHAIR APOSTOLAKIS: Okay. By the way, you tell me when a convenient point is to take a break.

MR. ARNDT: Okay.

CHAIR APOSTOLAKIS: You decide.

MR. ARNDT: Shortly.

CHAIR APOSTOLAKIS: Shortly.

MR. ARNDT: I've got about three or four more slides I want to do.

CHAIR APOSTOLAKIS: Okay.

MR. ARNDT: Briefly, the methodology is here. Since we've talked about a lot of this stuff, I will go through it real quickly. As we mentioned earlier, we developed a model of how the system works, state space model of how the system works. It can be anything you want. We're using a Markov model. You developed a statistical model associated with what you need to test based on different kind of failure states you have, how you do the modeling. You develop an operational profile; that is to say, the context of the system, what are the inputs, what are the different inputs it's going to see, what are the different interactions it's going to have? You construct a fault list based on how the system will interact and what potential failures you're going to have, back that through your model and come up with a list of potential faults you need to inject. You look at what is known as fault equivalents, which is a methodology to look at how the different input states would map to different output states, the same way you would do Latin Hypercube or various kinds of modeling methodologies to improve the statistics, a Monte Carlo calculation. You use that information to get for these systems the list of faults that you would need to do, you run the experiment, and you get the data.

CHAIR APOSTOLAKIS: So this is a design of a fault injection process.

MR. ARNDT: This is a design of a fault injection process.

MR. HICKEL: Let's clarify, when you say "a fault injection process", are you talking about faults that are -- where somebody corrupts maybe, let's say the set of stored constants, and then you let the thing do it?

MR. ARNDT: That would be one way to do it, yes.

MR. HICKEL: Or are you talking about

faults injected by simulating a failed sensor input, or both?

MR. ARNDT: Both.

MR. HICKEL: You're doing both.

MR. ARNDT: You look at all the different possible faults associated with the system. It could be failed inputs, it could be failed outputs, it could be corruptions, it could be software failures if you choose to do it that way.

CHAIR APOSTOLAKIS: But these don't necessarily have to be failures. I mean, I can select the values of the parameters that are extremely unlikely, and I can run the program. That's not part of fault injection. That's not a fault.

MR. ARNDT: No, that's not a fault.

CHAIR APOSTOLAKIS: It's a rare event.

MR. ARNDT: That's the operational profile. That's the space of inputs that's the system could possibly see.

CHAIR APOSTOLAKIS: Yes, I understand. But people do this as part of this --

MR. ARNDT: Yes. And the way you construct that is you look at both operational history, what has the system seen, and also what inputs will drive you to failures.

CHAIR APOSTOLAKIS: Now shouldn't there -- I'm sorry. Complete your thought.

MR. ARNDT: No, that's fine.

CHAIR APOSTOLAKIS: Okay. Shouldn't there be a statistical model there? It seems to me, one great challenge here is that there is a Box 8A or something that says we fix the faults. Yes, I mean, it's not that you are producing K failures and then trials, and then you go back and say well, now I'll do my Bayesian dance and so on. You fix those. So now what does that mean? Now what --

MR. HICKEL: Like George LaLuce and the rectification of ATWS 20 years ago.

CHAIR APOSTOLAKIS: Exactly. Exactly. Yes, sure. Yes, that's a similar thing. And the models I have seen out there, they are full of assumptions about these things, although this paper that was from the - I think it was from the IEEE - yes, "Transactions in Software Engineering" - that was a pretty serious paper, by the way.

MR. ARNDT: There's been some fairly significant work in this area. And the concept of fault injection goes back to the paper by Voso a number of years ago that looked at how this works. And there's been a lot of work in this area, and the idea is both to have a very high likelihood of uncovering failures, but also understanding them at a much level greater detail what that tells you about the future behavior of the system.

CHAIR APOSTOLAKIS: That's right.

MR. ARNDT: And that's what we're basically using it for in this application. Let me step through this, as basically the methodology that is used to go with that chart.

CHAIR APOSTOLAKIS: Yes, I think we discussed this.

MR. ARNDT: One of the big issues is the operational profile or the context. In our case, we're actually collecting data from the plant that we got the system from, as well as understanding the other possible assessments, and all that is at the control of the assessor.

This is just basically a chart that goes through and talks to the fact that we're not going to use a complete representation. We're going to break it down into modules or super components.

CHAIR APOSTOLAKIS: Yes, but this is where I got confused, as I said earlier. I mean, in Chapter 2, I thought you're presenting the system, the control laws and this and that.

MR. ARNDT: Right.

CHAIR APOSTOLAKIS: And then I saw this Markov thing, and confused -- there was a Chapter 4 in Markov.

MR. ARNDT: Right. Again, this is simply one way of representing the state space.

CHAIR APOSTOLAKIS: But are these rates that are produced in Chapter 2 used by Professor Aldemir in his Chapter 4?

MR. ARNDT: Yes.

CHAIR APOSTOLAKIS: So maybe you should move them then, because they are not used by DFM, I don't think. They are used by DFM?

MR. ARNDT: Yes. That's why it's structured this way.

CHAIR APOSTOLAKIS: Okay.

MR. ARNDT: We'll get to that after the break.

CHAIR APOSTOLAKIS: We'll get to that, yes.

MR. ARNDT: This is just a representation of how we put the various blocks together, the sensors, the main computers.

CHAIR APOSTOLAKIS: Well, this is it now. We have failure, or transition rates, or failures rates for each one of these boxes.

MR. ARNDT: We're going to have.

CHAIR APOSTOLAKIS: Well, that's what Chapter 2 does. Right?

MR. ARNDT: That's the methodology we're going to use to integrate the data we have with the testing we're going to do.

CHAIR APOSTOLAKIS: Yes. And, again, the issue of software problems is not covered by this picture.

MR. ARNDT: Let me -- this is one example of a state space diagram. They're functional states. You have an operational state, you have an operational state but with a loss of input, you have an operational state with a loss of output, you have an operational state that is unable to detect internal failures. Doesn't matter whether this is a hardware failure, rather hardware fault or software fault, or how the fault occurs in this particular methodology. It matters that the system goes from an operational state to a not operational state, or failed state based on some fault in the system. It doesn't matter in this particular model --

CHAIR APOSTOLAKIS: But, again, the question is, when you say "some fault", can you model all faults through the lambdas and the CIs. That's really the question.

MR. ARNDT: In theory, yes.

CHAIR APOSTOLAKIS: Well, but I'd like to see some discussion of that, a little deeper.

MR. ARNDT: Okay.

CHAIR APOSTOLAKIS: Why you can do that. And the CIs there, they really have a tremendous impact. I mean, the CI itself is .99, .999, so one minus that, you're talking about 10 to the minus 2, and 3, and so on. And, again, they have to be scrutinized why the number is .99.

MR. ARNDT: Right.

CHAIR APOSTOLAKIS: Okay. Good.

MR. ARNDT: And this is just the chart that you talked about. And at this point, we're going to talk about the PRA model and the actual modeling methodologies, and this is a good time for a break.

CHAIR APOSTOLAKIS: Very good. So we will reconvene at 10:25.

(Whereupon, the proceedings went off the record at 10:10:18 a.m. and went back on the record at 10:28:12 a.m.)

CHAIR APOSTOLAKIS: Okay. Let's go back in session. Steve.

MR. ARNDT: We're going to continue with Professor Aldemir talking about the PRA model and the DFM and Markov analysis, but before we start that, I thought it would be profitable for the Subcommittee to talk a couple of minutes about fault injection methodology; in particular, just to answer a few of the open questions from the last discussion. If this is not enough, we can have this as a separate topic at our next meeting. We'd probably want to do that, anyway. But while we're here, let's take five minutes and talk to a couple of the specific issues.

Carl Elks from the University of Virginia is here with us, and he will talk for a couple of minutes on that and answer your direct questions. Carl.

MR. ELKS: Okay. My name is Carl Elks from the University of Virginia. Just to give a little background, I started out doing fault injection experimentation and testing at NASA Langley Research Center in the early 90s, so I have some experienced based on this, along with modeling fault tolerant safety critical systems, and transitioning into formal methods at the University of Virginia, and also experimentation into safety critical systems.

The last discussion, we sort of talked

about conceptually what fault injection is, but I wanted to kind of just put a little finer point on some of the issues. Fault injection is a specific kind of testing regime to collect information out of the system to go into the models that we were talking about, specifically some of the Markov models, and even the dynamic flow-graph models. So the two parameters of interest to us as fault injection experimentalists are coverage, and we define coverage as the probability that an error detection mechanism or a fault detected given that a fault has occurred in the system is what we typically define as coverage. That is of importance to us because that also defines how well the system is responding to specific types of faults and fault classes.

Traditionally, fault injection has really addressed the issue of hardware-type faults, and other types of faults. There is work, and like Steve said, we're trying to transition this into the area of certain types of possibly design-type faults. That is certainly something that we are working with this committee to kind of address that. And more importantly, I think what Dr. Apostolakis said, that we really need to be mindful of, is we really need to state what the assumptions are behind all of the models that we're creating here, the data that is going into those models, and how that data has to instantiated into models to get credible results out of the system. And so one of the things that we have been doing at the University of Virginia is trying to develop a process by which these assumptions are explicitly stated. And we probably haven't done a great job of presenting that here today, but I wanted to state that that is a very, very important part of the research, to be very, very rigorous and scientific about how this information is generated, what assumptions are made there. And more importantly, can those assumptions be challenged and discharged with credible evidence.

CHAIR APOSTOLAKIS: Now the definition that is given in the report, for an example it says, "Say we inject 100 faults into the feedback loop, and we get two erroneous responses that were not detected by the system, then the non-coverage one minus C for that failure model is .02 ratio, and the coverage is .98." So the idea then of C is that you inject the number of faults addressing a specific potential failure mode?

MR. ELKS: That's correct.

CHAIR APOSTOLAKIS: Which you don't know

in advance.

MR. ELKS: Well, one of the things that Steve had me do early on in this project is to try to look at what I call generic failure mode taxonomy of INC systems, which would help us identify what are the important failure modes of this particular system, so that we could have some guided representation of exactly where to go into the system and inject these types of failures.

There are a number of different ways to conduct fault injection campaigns. One of them is what I call this guided fault injection. We're actually looking at particular hazards of the system that are either known, postulated, or some other theoretical method to say we need to look at this and go into the system and try to stimulate those and see what the responses are.

There's what I call the old school method, which is more random fault injection, where we statistically just go in and perform fault injections anywhere into the system and see what the response is. That type of fault injection is somewhat fruitless because you get a lot of non-responses out of the system, because you might be putting faults into spaces where the program is not executing. You might be putting it into spaces where there is actually no -- the timing and the actual data do not line up so that you'll get a response.

What we have tried to do at the University of Virginia is to use a combination of those two, based upon the information that comes from the system plant engineers who tell us, what is the most -- what do you worry about the most happening with this system? Give us your most dangerous fault list, so to speak. That's what I call it.

When I go in and talk to plant engineers or system engineers, I want them to give me this type of information so that I, as an experimentalist, and as a system analyst, can begin looking at the hardware/software interactions of the system to determine what types of things could go wrong to produce that most dangerous fault list.

CHAIR APOSTOLAKIS: Okay. If we pursue this example a little further, you inject the 100 faults.

MR. ELKS: Yes.

CHAIR APOSTOLAKIS: Ninety-eight of them, the system becomes aware of them. That's what you mean.

MR. ELKS: It's detected by the error

detection mechanisms.

CHAIR APOSTOLAKIS: How do I calculate this transition rate lambda?

MR. ELKS: You don't get transition rate lambda out of fault injection experiments.

CHAIR APOSTOLAKIS: Oh, okay.

MR. ELKS: What you get out -- you essentially get the coverage.

CHAIR APOSTOLAKIS: How do you get the transition rate?

MR. ELKS: The transition rate is input to the model. It really has nothing to do with the fault injection campaigns. The fault injection campaigns are strictly -- it's a stimulus response-type of testing-type thing. I'm trying to test the error detection mechanisms in the system to determine if they can detect certain types of faults.

CHAIR APOSTOLAKIS: So Table 2.4.1 then, the dependability parameters for the DFWCS system, where do these come from? I mean, I understand now where the Cs came from, where did the lambdas come from?

MR. ELKS: The lambdas come from, basically, talking with the plant engineers.

CHAIR APOSTOLAKIS: Oh, they're expert

opinions?

MR. ELKS: Collected on actual collected failure data rates, and also from the RAC Prism database of those two, so they're estimates based upon actual data, and actual database data.

CHAIR APOSTOLAKIS: It would be useful to see what data are used to produce this at some point.

MR. ELKS: This also opens up another issue. I think Dr. Apostolakis talked about this, was the viability of the failure rate data. I mean, these particular numbers that we have here come from both historical plant data, and out of a commercial database. It is known that these types of failure rates have a certain amount of uncertainty to them, because they're taken across a wide spectrum of applications, and everything like that. So when we typically do our analysis, either reliability or safety analysis, we do sensitivity analysis also with respect to some of these failure rates and coverage rates to see where the system is most sensitive to a particular failure rate, or a particular coverage rate, because that is also information that you can feed forward into the process to say this particular component has a failure rate, but if we vary that failure rate, the system reliability is impact greatest on this particular two parameters, so it's also a way of determining certain other aspects of the system that you just don't plug numbers into the model and get a number out. You kind of have to look at it in also in kind of what I would call a qualitative way.

CHAIR APOSTOLAKIS: So it seems to me that a very important question we have to address at some point is these lambdas.

MR. ELKS: Yes.

CHAIR APOSTOLAKIS: How they relate to what Brookhaven is doing, or other information, or whatever.

MR. ARNDT: We will at our next meeting have a specific session on data, both in terms of what's out there --

CHAIR APOSTOLAKIS: That mic is not working.

MR. ARNDT: We'll take as an action for our next meeting to have a specific session to talk about both what the data is out there, how we propose to use the data for our own internal independent validation methodology, and issues for the regulatory guide on data. And we'll talk about this, we'll talk about the more generic data work that Brookhaven is doing, and roll that in. You'll get some of that in the discussion later this afternoon, but we'll take an action to have a specific session on that next time.

CHAIR APOSTOLAKIS: Very good.

MR. ELKS: So I guess the last final thing I would like to say is this issue between the hardware/software interaction. The way that we inject faults into the system can be represented as some type of corruption of a register file and a microprocessor, or anything. And we typically represent that as kind of like some type of hardware failure in a microprocessor, and I'm using a microprocessor as an example here as something that we inject faults into.

In addition, we can also kind of represent - there's two ways to kind of represent sort of software-type failures, and those have to do with sort of like constructs that could be into the system that are activated by certain types of profiles that are going on in the system, as well. That's two different distinctions that we make. And the third thing that I would like to make is, is that as you're conducting this experiment, as I'm going through injecting errors into the system and everything like that, there's a very likely, and we've seen this at the University of Virginia, and I've seen it at NASA - it's very likely that you find that an error detection mechanism or some other component of the system behaves in a way that it wasn't intended. It's a specification error, it's a design error at that point in time. And we look at it and we go oh, okay. This is a true bug into the system. So the technique addresses both types of faults, but in a legacy sense, it originally started out as hardware and has since transitioned in to represent these hardware/software-type interaction faults, as well.

CHAIR APOSTOLAKIS: Great. Thank you. So this is an action item for the future.

MR. ALDEMIR: Well, what I'm going to start talking about is the example PRA model that we have adopted. And the reason for adopting a PRA model is that eventually we would like to quantify the effects of digital versus analog, or the effect of switching over to a digital system on the overall core malfrequency and the large early release frequencies. The plant we chose is a NUREG 11.50 plant. It's a three-loop design, and we are assuming that the control system is applicable to each of the loops.

So the example, the event that is used in this report that was being referred to is a turbine three trip event. We talked about it earlier. This is the conventional event tree analysis of the event, and since everybody is familiar with this procedure here and the events, I'm not going to go through that. But basically, we tried to keep the water level in the steam generator using the oscillator feedwater system. If it doesn't work, then we switch over to main feedwater system, making the turbine active, and then you have another number of sequence of events following that, which are not going to be all that much relevant to our example. This is the rest of the turbine event tree, and as I said, as far as our control system is concerned, we are not so much concerned with this part of the event tree.

Now the methodologies we have identified earlier, and these were among the conclusions of NUREG 69.01, is that the dynamic flow-graph methodology and Markov methodology, and as distinct from what has been discussed earlier with respect to Chapter 2 of this report, that is a methodology to decide what sort of faults to inject, and where to inject them. This Markov methodology is to predict system reliability, or rather, is a reliability model of the system, and it needs input from the earlier discussion of data generation.

The first methodology, dynamic flow-graph methodology, was developed by ASCA in the early 90s to support risk assessment. The software was used in safety analysis of several software control systems, and the results validated dynamic flow-graph's methodologies, ability to handle software/hardware interactions, and to perform dynamic analyses, specific applications, digital feedwater control system in a pressurized water reactor which was published as NUREG/CR 6465, control system for the combustion module, one system of a shuttle experiment.

The important features, graphic modeling environment and automated analysis engine that can handle cause/effect relationships, time-dependent relationships, feedback loops, the state vectors represent key process parameters, and mapping between the state vectors governed by multi-rated logic rules which are represented through decision tables, transfer boxes, transition boxes in the graphical mode. And we'll see examples of these in a little while.

Once you construct the DFM dynamic flow-graph model, you can either analyze it inductively or deductively. Now in the inductive mode, it's the forward-tracking/discrete-event-simulation mode, you are trying to identify the possible combination of components failures, even initiating event, and deductively you are going backwards and given the undesirable event you are trying to identify what sort of event sequences have caused it. And you can interrogate the dynamic flow-graph methodology model several different ways, and again, as I indicated, deductively/inductively. And also, there is another mode that will come later on that will allow you to decide what type of testing you can perform.

In the deductive mode, the software identifies prime implicants, and these are distinct from minimal cut sets in the sense that they are multi-valued logical equivalent of minimal cut sets. And, particularly, they become important when you have the events - the importance of time-dependence of events, like the example I told you. In fact, we have identified - when I say we, I mean ASCA has identified these two different failure modes that differed by a second or so by using dynamic flow-graph methodology, and I'll come to that in a little while.

This is a fairly standard approach.

CHAIR APOSTOLAKIS: The first bullet is interesting. Do you have probabilities for all the events that appear in the prime implicants? That's multi-valued, right?

MR. ALDEMIR: Well, the prime implicants will depend on what sort of basic event, so to speak, we have considered, what sort of failure modes, what the state space consists of. So if we have data for the state space, this will feed input -- this will feed into the DFM. So basically, lambda times Delta T, since we are doing discrete-event-simulation, is going to give you those probabilities, the lambdas that we talked about earlier times the time increment.

CHAIR APOSTOLAKIS: They don't rely on transition rates here, do they?

MR. ALDEMIR: Well, in the quantification process -- well, DFM you can use in different modes. You can use it for qualitative analysis, get the prime implicant, or you can quantify the prime implicants, and they --

CHAIR APOSTOLAKIS: Then these will have events such as this parameter is between A and B. And there is a probability that that parameter is there. Then at the next step, there is a transition probability that a parameter moves to another interval? That's where I get lost.

MR. ALDEMIR: Well, we are not -- okay. That would be the initiating event, distribution. Now if we're talking about - if the system states include parameter values being in certain intervals, and are you referring to the dynamics of it, or are you referring to the modeling parameters?

CHAIR APOSTOLAKIS: All the parameters are selected at the beginning.

MR. ALDEMIR: Okay. So we're talking about the modeling parameters --

CHAIR APOSTOLAKIS: Yes. Yes.

MR. ALDEMIR: -- that represent the dynamics. At this point, neither of these methodologies - well, I have to clarify that later on - Markov does it a little bit the way I'm going to define it, but that is not our emphasis in the modeling. We're assuming that those are given. Now what would happen if they change would be the subject matter of a sensitivity analysis.

CHAIR APOSTOLAKIS: At some point it would be useful to try to relate the prime implicants to the states that you have in the Markov model.

MR. ALDEMIR: Actually, what we are planning to do is compare the prime implicants -- actually, you will see in a little while that both Markov methodologies, and I'm referring to the one in Chapter 4 of the report, and DFM, are pretty much the same thing. We can produce, the results of exchangeable. One has some advantage in a certain area, and the other one has advantage in a certain area, but we are doing pretty much the same thing. In fact, what we are planning to do is to generate prime -- Markov can generate prime implicants, as well. So we will generate independently these prime implicants, compare them, and resolve the differences. That's one of the exercises that we are planning to do. We have already done it in a partial way, but since we are doing this independently purposefully so that we don't influence each other, we have assumed different initial conditions.

CHAIR APOSTOLAKIS: Does the Markov model use multi-valued logic?

MR. ALDEMIR: Yes.

CHAIR APOSTOLAKIS: So you will have a chapter at some point in the future where you will do these things?

MR. ALDEMIR: In this report, we'll -- okay. The report is out for review right now, and it will be revised, depending upon the reviewer comments. And if this is a point that they also would like to see, it's a matter of also timing issues. If there's time, we will put this comparison in this one. It's a matter of timing, actually, the deadlines. It's a matter of doing some of the analysis again.

Now if there is no time to do it for this report, what we will definitely do is for the next report, where we will quantify what qualitative comparison and quantitative comparison, and resolve the differences.

CHAIR APOSTOLAKIS: Maybe it would be wise to include that comparison in this report, because if this report is issued separately, then people may assume that either methodology is fine, and the NRC published it, so we can do it.

MR. ALDEMIR: Oh, I see what you're saying.

CHAIR APOSTOLAKIS: But if you have a comparison.

MR. ALDEMIR: Good point.

CHAIR APOSTOLAKIS: And also, a critical evaluation of the rates. I think these things go together.

MR. ARNDT: Yes. The idea is that this is a staged approach. We looked at the various methodologies that might be appropriate, we chose a few that we thought would capture the characteristics we were interested in, and how they could be constructed, which is the purpose of this report. And then the next report will be how well those systems actually work in doing these kinds of analyses.

CHAIR APOSTOLAKIS: What's the rush for publishing this one?

MR. ARNDT: There's no rush. The point is, before we go forward with the regulatory guide saying these are ways that we think are acceptable, and it's nice to be able to point to a document that is in the public domain to articulate that.

CHAIR APOSTOLAKIS: But it seems to me that you will be in a better position to define what's acceptable if you do this comparison. Bill?

MR. KEMPER: Yes. Bill Kemper, again. Thank you. Steve has kind of hit the nail on the head here. We're really under internal pressure of our own to try to move on with this and get some regulatory guidance out there as soon as we can, because we think the industry really is desirous of this. This series of NUREGs, as Steve said, will provide the underpinning or the regulatory bases, if you will, for the Reg Guide itself. And also, we have an industry public meeting coming up in August, which we've had to slip a couple of times, and I'm hoping dearly that we don't have to slip it again, so this plays into that, as well. We want to have as much information out there available to the public before that public meeting.

CHAIR APOSTOLAKIS: I still think, though, that the critical evaluation of the failure rates and position rates should be in this report.

MR. KEMPER: Well, what we can do is we'll look at the time implications of that, and if we can do it, Tunc, Steve, do you all see any reason not to do that? I mean, assuming that it doesn't completely washout our schedule here, obviously.

MR. ARNDT: The intention is all of these issues will be covered by the time we finish with third report. It's just a matter of which report and what the exact timing is, and whether or not it becomes logistically challenging to publish this report with that information that may delay it so far that it makes no sense to publish the third report. There's logistical issues here, as well.

CHAIR APOSTOLAKIS: But if the source of doubt regarding the applicability of Markov systems is this meaning of the REGS, it seems to me you should address it. I'm not asking for a major treatise, but you should address it in the report, and acknowledge that there is this issue, and here is our answer.

MR. ARNDT: We certainly need to

acknowledge that it's an open technical issue, and this is how we are choosing to work it, and this is why.

CHAIR APOSTOLAKIS: So are you saying that the regulatory guide will refer to these methods?

MR. ARNDT: It will reference this as information, but as we've talked about about four times already, there is going to be some systems that don't need this sophisticated modeling, so that part of it will reference other sections. But the information we've learned in developing this information is something that we want to use as a technical basis for the decisions that we have in the regulatory guide. If we say that there are some systems that need this level of modeling, then we need to point to both open literature and NRC literature that says this is what our issues are.

CHAIR APOSTOLAKIS: Well, I mean, I appreciate the issue of schedule and all that, but I mean, certain things are really important.

MR. ARNDT: We appreciate the --

CHAIR APOSTOLAKIS: Do we comment on NUREG reports? We do.

MR. THORNSBURY: Some.

MR. KEMPER: You can, but generally we

don't ask that you do that.

CHAIR APOSTOLAKIS: But we can volunteer.

MR. KEMPER: You certainly can.

MR. THORNSBURY: You're a member of the public, too, George.

MR. KEMPER: This is true, you are a member of the public. Well, I think Steve's point here is we will do what we can to address that and move forward, try to preserve our schedule commitments as best we can.

MR. ALDEMIR: We will also try to see if we can have at least a qualitative comparison of the prime implicants that we get from Markov and DFM. That was already in the --

CHAIR APOSTOLAKIS: It's fine to have something and then say more details will be somewhere else.

MR. ALDEMIR: No, I think we have --

CHAIR APOSTOLAKIS: But not to say anything is not really acceptable.

MR. ALDEMIR: If we are using the same scenario to simulate it, it only stands to reason that we compare the results, and try to resolve as many difference as possible. It may not be possible to resolve all of them, in which case we'll then defer to the third --

CHAIR APOSTOLAKIS: By the way, I think it needs a good editing job, this report.

MR. ARNDT: Yes. The version that you got was a very early version. We wanted to provide you the information for your technical background.

MR. ALDEMIR: The DFM analysis --

CHAIR APOSTOLAKIS: Say you have an actual replication of this? Are you going to show the actual --

MR. ALDEMIR: Yes. You want me to skip all this?

CHAIR APOSTOLAKIS: Maybe you can go there.

MR. ALDEMIR: Okay.

CHAIR APOSTOLAKIS: Because I don't think this means anything to anybody who is not familiar with the method.

MR. ALDEMIR: Okay. Let me first do kind of -- anticipate where we are going, and as I said in the beginning of my presentation, that we will eventually need to integrate these models into an existing PRA. So this is one way you can do the integration, and we are using SAPHIRE as the tool, and the turbine trip event as the initiating event. You can, in the graphical mode, you can simply graphically insert these types of –- the event sequences that have been obtained through prime implicants into the event tree. Then I will show you later on, and we illustrated this for Markov - I'm sorry, the dynamic flow-graph methodology, and then for Markov I will use another mode of SAPHIRE input to show how we can include them –- incorporate them into SAPHIRE. But both methodologies can be used in both modes.

So example initiating event –-

CHAIR APOSTOLAKIS: Let me –- let's go back one second. This is a static representation of the system.

MR. ALDEMIR: Right.

CHAIR APOSTOLAKIS: And you are doing a dynamic analysis. So how am I to interpret the event MFW phase? When?

MR. ALDEMIR: Okay.

CHAIR APOSTOLAKIS: Are you going to give me a global event or what?

MR. ALDEMIR: In this particular –- that's a very valid point. In this particular illustration, the timing doesn't matter. The event sequences, I mean, the prime implicants, the timing is not an issue here. So if that's not an issue, then we can take it and simply incorporate it statically into a fault tree.

CHAIR APOSTOLAKIS: It's not an issue?

MR. ALDEMIR: In this particular example that we're talking about.

CHAIR APOSTOLAKIS: So why are we using dynamic --

MR. ALDEMIR: No, no. We chose an initiating event, example initiating event. Now in this situation, we have two types of responses, either the system behaves and fails in one mode versus the other. So we get the prime implicants that lead to these events. Now there are - I forgot the number, but there are about 11 implicants, prime implicants that lead to one type of failure, and then five, six, or seven that lead to the other. We conglomerate them so you have top event failure - I mean, sorry - high level or low level.

Now, again, coming back to why are we doing this dynamically? Well, you may be able to identify the faults, I mean, the failure modes. And, in fact, you have to specify them up front what sort of failure modes you're going to have. The question is, when you start quantifying them, unless you take the dynamics into account, you may get different results.

CHAIR APOSTOLAKIS: But then how far you will go into time? I mean, this still says failure of the main feedwater --

MR. ALDEMIR: These are all valid issues. These are --

CHAIR APOSTOLAKIS: Are you going to say I'm going to 100 seconds, 50 seconds?

MR. ALDEMIR: These are all valid --

CHAIR APOSTOLAKIS: Is it possible that you may even create another branch?

MR. ALDEMIR: These are all valid issues.

CHAIR APOSTOLAKIS: So we haven't resolved those yet.

MR. ALDEMIR: No.

CHAIR APOSTOLAKIS: Okay.

MR. ALDEMIR: In fact, some of them are not resolvable.

CHAIR APOSTOLAKIS: Whoa, whoa. We're not squaring the circle here.

MR. ALDEMIR: Well, the issue is the following. If you have an existing PRA based on a static model, you generate the dynamic model. All these issues that you brought up are valid. Well, then you have to make certain assumptions. For example, you look at the event tree and they say how was this generated? What was my assumption on the initiating event here? And then you go back to your dynamic model and use the same initiating event, then things will match.

CHAIR APOSTOLAKIS: But you will address this some time in the future.

MR. ALDEMIR: That's why we are doing it in the third report. That's why --

CHAIR APOSTOLAKIS: Well, that's the thing, again. I mean, if you issue this report and a guy tries to make some real life decisions using this as a basis, and then this question comes to his or her mind, I mean, how useful is the report? I mean, there are important issues that have to be addressed.

MR. ALDEMIR: Again, we are assuming that the existing PRA does not change, we cannot change that, so the question is how can we fit it best into the existing PRA. So one way - and all these issues that you brought up are relevant, so then we look at how the original PRA was constructed, and try to make the same assumptions in our representation.

CHAIR APOSTOLAKIS: Will you at least have in your conclusion section a discussion of these issues, without necessarily giving an answer?

MR. ALDEMIR: Yes, sure.

CHAIR APOSTOLAKIS: Because, you know, a user will feel much better if he appreciates or he realizes that the authors of the report appreciated these issues.

MR. ALDEMIR: As I expressed, how far you are going to go, same thing with the event tree - I mean, you come to a stop when you reach a consequence of interest to you, and the same thing you can do this. You can do it for the dynamic methodologies, you can follow them as far as the events in the event tree go.

CHAIR APOSTOLAKIS: That's one approach.

MR. ALDEMIR: Yes, I mean that's one way.

CHAIR APOSTOLAKIS: That makes sense.

MR. ALDEMIR: But a key issue here is, when you are tying up these links, am I making the same assumptions in the linkage. And then you have to see what the initial assumptions were in the event tree generation so that you generate your dynamic methodology or dynamic event tree the same way. And, of course, you may need to –- if you have no information, what if you have no information? Then you do a sensitivity analysis on the initial conditions, try to see how much of a difference it will make as far as consequences and event development goes, as to what assumptions you make in initial events. But this is what we will defer to as epistemic uncertainty.

CHAIR APOSTOLAKIS: Yes. Everybody refers to it. Another thought occurred to me - there was a question last time you guys were - I mean, Steve was before the Full Committee - there was a question from a member, or a comment, that universities really produce methods and ideas and all that, but then there is this extra step of making something operational, where you need now the regulatory guides, guys, or National Laboratory to take over and make it practical. And, Steve, you said yes, that we are at the stage we're producing ideas and methods, and there will be a second step. But today, I get the impression that you're going into regulatory guide directly, without having this intermediate step, where somebody actually uses these, trying to make it --

MR. ARNDT: We're going to talk a little --

CHAIR APOSTOLAKIS: -- say "practical".

MR. ARNDT: We're going to talk a little bit about that later in the afternoon. There's three things you need to understand.

CHAIR APOSTOLAKIS: There's a lot of things I need to understand.

MR. ARNDT: From a structural standpoint.

CHAIR APOSTOLAKIS: Okay.

MR. ARNDT: We go back to my bubble chart. One of the issues is developing a practical independent assessment methodology for the NRC. In that case, let's talk about that for 30 seconds. We come up with the ideas, we look at the limitations, we look at the advantages and disadvantages of various methodologies, we look at the data, we come up with an idea, then we transition that to the people who do this for practical day-to-day basis, in our case, the INL lab that runs the SAPHIRE and SPAR program. That is part of the plan for that part of the program. And we'll actually talk about that briefly today.

The other part is the development of guidance as to what we consider to be acceptable for review that the industry can bring in. We can do that in one of two ways. We can develop it and say this is an acceptable methodology, and go through all the gory details of what we think is acceptable or not, or we can write basically a performance-based regulatory guide that says we don't care what methodology you use, so long as it meets certain criteria.

At this point, we're planning on going down the second path, rather than the first path, for a number of reasons. One, because there's a lot of different ways to do this. We're looking at three, the traditional fault tree/event tree methodologies, the DFM and the Markov. There are others. We have different characteristics, different aspects of that. The work that we are doing to develop our own independent assessment methodology will inform the development of our regulatory guidance, and we will point to some of that information as reasons why we make particular decisions in our regulatory guidance.

CHAIR APOSTOLAKIS: Okay. It will be exciting when we review the regulatory guide.

MR. ARNDT: For a whole bunch of people.

CHAIR APOSTOLAKIS: I can see people getting very enthusiastic when you tell them find the prime implicants.

MR. ALDEMIR: Do you want me to go through the DFM model construction procedure? The idea is --

CHAIR APOSTOLAKIS: Well, keep going. I don't know. We will stop you when we think --

MR. ALDEMIR: Okay. The idea is basically a graph theory oriented approach. We take the discretized process parameters as nodes, we represent them as nodes, and we have transfer function between the nodes expressed as decision tables. So in this chart, which corresponds to what I have described as the example initiating event, it's DFM modeling of the same event sequence, or the system, the part of the system that involves that event sequence.

CHAIR APOSTOLAKIS: So where are the control laws in this --

MR. ALDEMIR: Controls laws are going to be going through the transfer boxes. It's going to be represented as the decision tables --

CHAIR APOSTOLAKIS: Easy to develop decision tables using control laws.

MR. ALDEMIR: Now, my understanding is, actually, we can ask Mike --

CHAIR APOSTOLAKIS: Mike is here. Right?

MR. ALDEMIR: Why don't you come and explain?

CHAIR APOSTOLAKIS: Identify yourself.

MR. YAU: Michael Yau, ASCA, Incorporated. To answer Professor Apostolakis' first question regarding the control laws, the key parameters in the control logic are the ones highlighted inside the green brackets.

CHAIR APOSTOLAKIS: Okay, on the left.

MR. YAU: On the left. That's right.

CHAIR APOSTOLAKIS: So am I to understand there is a decision table behind each of these symbols there?

MR. YAU: Right.

CHAIR APOSTOLAKIS: And then you did what? How did you develop these? I mean, you solved the equations?

MR. YAU: Basically, I -- in the control law translated into a software sub-routine, I supplied a range of inputs for the sub-routine, and then from the outputs, look at the outputs and then build the decision tables from the relationship between the inputs and the outputs.

CHAIR APOSTOLAKIS: And time comes into this? I mean, the decision table, again, is a static representation.

MR. YAU: Not necessarily. Decision table can be a dynamic representation in the sense that you supply the inputs at a time step before, and then you get the outputs a time step later.

CHAIR APOSTOLAKIS: And that's time independent? You see what I'm saying? No, it can't be.

MR. ALDEMIR: It could be time

independent, if the system --

CHAIR APOSTOLAKIS: Could be, but --

MR. ALDEMIR: If the system is autonomous, yes. If it is not, then they create another decision table, basically.

CHAIR APOSTOLAKIS: And what's the time step here, Mike?

MR. YAU: Right now in the model that we are putting, it's assumed we are running -- the decision tables were built based on time step of 10 clock cycles.

MR. ALDEMIR: In this example, the system is not autonomous because the decay -- the heat generation rate is an exclusive function of time, so the decision tables will have to be built as a function of time.

CHAIR APOSTOLAKIS: Have they been built that way? I mean, that's an important point. I mean --

MR. ALDEMIR: Yes.

CHAIR APOSTOLAKIS: They have.

MR. ALDEMIR: Well, Michael will help me out, but this --

MR. YAU: Well, actually the decay heat part is really part of the input to the software.

It's the compensated power, and we -- in the input used to generate the decision table, we sample a range of the input power from zero percent to 100 percent, so you have the representation, if the power is in this range, we've got these set of outputs. If the power is in a different range --

MR. ALDEMIR: They are basically converting to the autonomous system in this situation.

CHAIR APOSTOLAKIS: Okay.

MR. ALDEMIR: So the decision table will be static. But you can do it dynamically, so it's just a matter of depending upon how the system representation is.

MR. ARNDT: The real point here is the level of detail you need in the model, be it this model or any other, is dependent upon the amount of the features of the system that you need to capture for it to be an appropriately representative model. So, for example, when we talked about the aspects of the model, the watchdog timer, if the main computer has a fault, it'll shift to the backup computer. That's a time sequence. There's issues associated with the characteristics of the system, so the amount of timing you have and the amount of detail you have is based on the amount -- the feature of the system you want to capture.

CHAIR APOSTOLAKIS: Yes, but at the same time, if by capturing those features you come up with a methodology that is completely unmanageable --

MR. ARNDT: Well, that's the point of doing the study, to see whether or not you can do that.

CHAIR APOSTOLAKIS: So this was manageable, Michael?

MR. YAU: For this simplified benchmark system, it is. But let's say if you have a more complicated software module that models a common filter, I don't think you can do a practical decision table that way. I think you have to rely on some clever method of dividing the input space into different contexts, and then rely on testing to build the decision table.

CHAIR APOSTOLAKIS: I see. There's a way around.

MR. YAU: There's a way around, yes, sir.

CHAIR APOSTOLAKIS: Okay. Let's go on.

MR. ALDEMIR: Since you are here, why don't you step through these.

MR. YAU: So, basically, from the DFM model that was constructed to represent the feedwater control system and the steam generator, we could analyze this model for different top events. The two top events of interest are the steam generator at a high level, and the steam generator at a low level. These top events were defined as a conjunction of the state of the knocks represented by the DFM model, and the top event that this third bullet corresponds to is the high level top event.

The level was discretized into five states, two, one, zero, negative one, and negative two; two being the highest, and negative two being the lowest. What this top event says is that I want to find out what are the prime implicants that could lead me to the highest level at time zero, while passing through level one at time T minus 1, and starting from the normal level at T minus 2. Given that the ELP and the CZL variables are zero, that means you don't accumulate a lot of errors inside the PID control logic. There are not a lot of integral errors in the control logic, so you're basically starting from a very nominal state, and then somehow progress to the high level. And then the DFM model was analyzed deductively for two time steps for that top event, and the 11 prime implicants were identified.

CHAIR APOSTOLAKIS: So this is now for

what time, time zero? The 11 prime implicants at which time?

MR. YAU: At time minus two. We were backtracking two time steps, so our top event occurs at time zero. But we find out things that happen before --

CHAIR APOSTOLAKIS: You go back two times, yes.

MR. YAU: Right. Before.

CHAIR APOSTOLAKIS: So 11 prime implicants for time zero.

MR. YAU: Right.

CHAIR APOSTOLAKIS: Right.

MR. YAU: And then --

CHAIR APOSTOLAKIS: And did you guys find this 44 second --

MR. YAU: No. Actually -- the fact is that these prime implicants, they don't tell you exactly okay, this thing happens at 44 seconds. It just gives you the initial condition, and one of those initial conditions corresponds to the 44 second case. Let's say we focus on prime implicant number 5, it says the level was normal at time T minus 2, the level error is nominal, the compensated level is nominal. But then at that moment, the feed flow is greater than the steam flow, and then your bypass flow valve failed stuck. And that's the condition at 44 seconds, because at that moment feed flow is greater than steam flow, and if your bypass flow valve got stuck, then the feed flow/steam flow mismatch will lead you to a high level. That's basically what the prime implicant tells you. It doesn't tell you that you have to look specifically at 44 seconds, but you have to look for cases where the steam flow and the feed flow mismatch, and then you can have a stuck position.

CHAIR APOSTOLAKIS: Now you report the probability here of 2.5 ten to the minus 4, not there, in the report.

MR. YAU: We removed those, because basically those numbers were assumed numbers, and we subsequently removed those.

CHAIR APOSTOLAKIS: All right. I was trying to find out why they're in the --

MR. ALDEMIR: No, we removed those numbers.

MEMBER BONACA: Forget it now.

MR. YAU: Those numbers are basically used to illustrate how you could go from the prime --

CHAIR APOSTOLAKIS: Okay. Let's say you want to quantify this, again, prime implicant five, level is normal at T minus 2. That's one, right. I mean that's -- yes, really normal is one.

MR. YAU: I think you could get those numbers from the operational profile. The level may be --

CHAIR APOSTOLAKIS: A very high probability of --

MR. YAU: Yes, that's right.

CHAIR APOSTOLAKIS: Level error is nominal.

MR. YAU: It comes from the operational profile in the software. Basically, you accumulated a very small error, and you can easily correct this.

CHAIR APOSTOLAKIS: You can have a probability for that?

MR. YAU: I don't know how to generate that, at the moment.

CHAIR APOSTOLAKIS: Ahh, okay. Compensated level is nominal. Tunc, you want to say something?

MR. ALDEMIR: These are initial conditions, basically.

CHAIR APOSTOLAKIS: All of these are initial -- yes, but --

MR. ALDEMIR: Blue are initial conditions.

CHAIR APOSTOLAKIS: Why are they initial?

MR. ALDEMIR: Because you have third order system, you need three initial conditions.

CHAIR APOSTOLAKIS: If it goes back two steps. Okay, fine. But still -- okay, so these are -- feed flow greater than steam flow. That's red, right? So that's not an initial condition. So how would you get that probability?

MR. YAU: We don't have an answer right now, but I would venture to speculate that you would try to quantify it by looking at the operational profile and see how the steam flow and feed flow profile under this initial condition.

CHAIR APOSTOLAKIS: So we do have some issue here how to get those probabilities, so the main value of this is the qualitative --

MR. YAU: Qualitative at the moment. That's right.

CHAIR APOSTOLAKIS: What it takes, what kind of states it takes to lead to the undesirable event.

MR. YAU: Right. As Professor Apostolakis pointed out earlier, from this qualitative analysis, you might want to really fix these kind of issues before even you try to quantify them. You may want to have some check --

CHAIR APOSTOLAKIS: And then you have the same problem like everybody else.

MR. YAU: That's right.

CHAIR APOSTOLAKIS: The only thing you can do is just assume some rates. If other people can do it, you can do it.

MR. ALDEMIR: Again, they had such -- how you would get the number, operational profile, you need some input data, like in any other initial event --

CHAIR APOSTOLAKIS: Well, what do you mean by "operational profile"?

MR. ALDEMIR: How many times have you observed this kind of event.

CHAIR APOSTOLAKIS: At T minus 2, zero.

MR. ALDEMIR: No, no. No, no.

CHAIR APOSTOLAKIS: Oh, come on.

MR. ALDEMIR: How many times have you observed feedwater being - what is it - feed flow being larger than steam flow? The minus 2 is not relevant here. It's just the probable distribution that's relevant.

CHAIR APOSTOLAKIS: I don't know. We'll have to think about that. That's certainly an input to it.

MR. ALDEMIR: But, I mean, you would definitely need inputs. Again, the dynamic analysis, like any other -- even with normal event tree efforts, you would still need to observe or know how system will behave as a function of time --

CHAIR APOSTOLAKIS: I understand that, and I think right now, I think that the greatest value of what you guys are doing is qualitative. That's my view. And the jury is out whether the quantitative information is realistic and practical. That's my view. Two guys nod, two refuses to -- that's fine. That's fine.

MR. ALDEMIR: If I start responding, this is going to get into a more philosophical mode. In any kind of engineering field, we do the best we can.

CHAIR APOSTOLAKIS: Oh, don't -- yes, okay. Let's go on.

MR. ALDEMIR: I mean, we cannot say wait, we don't have anything.

CHAIR APOSTOLAKIS: I understand.

MR. ALDEMIR: Okay. Should I go through these fast, or are we --

MR. YAU: Actually, I could just skip through them really quickly.

MR. ALDEMIR: Well, you might as well say a few words.

CHAIR APOSTOLAKIS: Let me understand this T equals zero. So this is the actual start of the transient, the zero, or is it your zero?

MR. YAU: My zero. It's not the start of the transient.

CHAIR APOSTOLAKIS: It could be any time, actually.

MR. YAU: Right.

CHAIR APOSTOLAKIS: Okay.

MR. YAU: Basically, what I'm saying is that my top even time is this zero.

CHAIR APOSTOLAKIS: I understand. Why did you choose to go back only two time steps, and not three?

MR. YAU: Because in the simplified model, I know that the level could go from zero to two in two time steps, so that's the minimum number of time steps required to get there.

CHAIR APOSTOLAKIS: I see. So there's some logic.

MR. YAU: Right.

CHAIR APOSTOLAKIS: Okay. That's good.

MR. ALDEMIR: Should I --

CHAIR APOSTOLAKIS: Yes, let's skip now. Remember, you have to finish at 12:00.

MR. ALDEMIR: I know. It's going to be hard. Well, I will just then try to go through the Markov methodology fairly fast. But before we start, this is, again, a way to predict the system reliability, so it's a predictive model. And what we are using earlier was a kind of an inductive model to figure out what kind of inputs, what kind of faults we're supposed to be injecting, so these things are totally disassociated, except that the former model, the one that is used for fault injection, helps to feed data into this model or DFM.

CHAIR APOSTOLAKIS: The discussion we just had, with DFM, Mike produced the prime implicants, which are qualitative insights into the system without using any quantitative information. Can the Markov model produce qualitative results without failure rate numbers?

MR. ALDEMIR: I'll show you. I'll show you in a little while. It does. This is a recent development, incidentally; developed as part of another project. So in the Markov methodology, we --

CHAIR APOSTOLAKIS: Why do you call it Markov?

MR. ALDEMIR: Because it's a Markov model. I mean, the main -- we discussed this with other member of ASCA, and the main difference between two methodologies is, in the decision tables they assume zero one, we assume non-zero values, as well, non-zero/non-one, we're in-between, as well. That's the only difference.

CHAIR APOSTOLAKIS: But the problem -- what I don't understand is this. In the Markov model, you start with a Markov diagram, which you build. Correct? The states.

MR. ALDEMIR: Yes. But the same states go into DFM, too. They have to --

CHAIR APOSTOLAKIS: Well, in there is the truth tables?

MR. ALDEMIR: Well, you need to have some certain states of the system so that you can figure out what possible -- to construct your decision tables, you need --

CHAIR APOSTOLAKIS: Well, I really think you need a closing chapter with some of these things.

MR. ALDEMIR: As I said, we will do comparisons. Now it is going to be difficult to relate one to one, because then the report is going to become unmanageable, because if you look at the report, we don't have too much on DFM because it's already been out there. There's one NUREG already published on it, 64.65.

CHAIR APOSTOLAKIS: Sure, sure, sure. But some comparison, I think, would be useful on the basic stuff. Yes, you see the experienced guy. Say yes.

MR. ALDEMIR: Okay. Yes.

CHAIR APOSTOLAKIS: But we are experienced, too. We'll hold you to it.

MR. ALDEMIR: Okay.

CHAIR APOSTOLAKIS: You know, at this time maybe going to details like cell-to-cell and all that probably doesn't serve much of a purpose, so if you can give us the flavor of the approach, because you'll never finish, otherwise.

MR. ALDEMIR: Right. Okay. Let me then give you the flavor of the approach, what I just said earlier. I'll skip through these probabilities. So this is going to be something –- sorry, go ahead.

CHAIR APOSTOLAKIS: The equations, the control laws, how do you use them in the Markov model?

MR. ALDEMIR: As I said, the only distinction between us - I mean not us - between Markov methodology and DFM is how we construct the decision tables. In our approach, in the DFM methodology, we use to one-to-one mapping, and correct me if I'm wrong, Mike - one-to-one mapping, so it's always zero or one. You still partition the process variables into ranges, and then you take one point from one end table, try to see where it will go following the system equations in a given specified time.

DFM uses one way, not because it's not capable of using more than one, it's just that the model becomes unmanageable. So in the Markov approach, the same philosophy, except using more than one point to start from each partition to map into each partition, to other partitions. So when the decision tables of DFM are zeroes and ones, Markov produces decision tables which may have values in-between. So that's the example that I was going -- this is kind of showing you how the mapping scheme is done. This is our representation of the transitions between each component state, between component states. These go as inputs into the Markov model. This is how you would construct these transition probabilities from process variable --

CHAIR APOSTOLAKIS: Your cell-to-cell --

MR. ALDEMIR: Cell-to-cell mapping, that's correct. This is the kind of decision table --

CHAIR APOSTOLAKIS: Go back one. I remember in the report you say somewhere that some of these factors can be obtained from look-up tables, or am I - I don't remember correctly?

MR. ALDEMIR: It depends on the complexity of the system. If the system –- the equations describing the system dynamics is a convenient way of –- well, one way of system modeling. You may actually use look-up tables if you have experimental data on system performance. Say that the system performance is not that complicated, and you have –- let's say you know that if I am in this interval, I will be in that other interval based on experimental data, based on observation, based on expert judgment, if you want to.

CHAIR APOSTOLAKIS: Otherwise, you produce it?

MR. ALDEMIR: Otherwise, you can produce them through the –- I mean, you just need a system model, whether it be qualitative, quantitative, doesn't really matter, integral, differential equation, as long as you can map one time step to the other time step, and both methodologies do the same thing, both DFM and Markov do the same thing.

CHAIR APOSTOLAKIS: All right. Let's go on.

MR. ALDEMIR: This is the kind of decision table that you will build, and from what I understand, DFM does pretty much the same thing. The differences you see are here. These are not all zeroes and ones. There are probabilities associated with these transitions. And it's not because DFM cannot do it, it's just that the model becomes very complicated. They choose usually not to do it.

CHAIR APOSTOLAKIS: This is the kind of thing that would be nice to explain a little bit in the report. I really think it would go a long way --

MR. ALDEMIR: The similarities, we --

CHAIR APOSTOLAKIS: Similarity, why you have .33 and they don't. I mean, it's not a big deal.

MR. ALDEMIR: Sure, sure. No, there's no problem with that, no.

CHAIR APOSTOLAKIS: Within half an hour, can't you --

MR. ALDEMIR: No, no, no. Actually, as I said, we are planning to do --

CHAIR APOSTOLAKIS: No, refer to that you cannot do it, or what? It cannot be done?

MR. ALDEMIR: No, we will do it. We were planning to do it, as I said, after the -- we are waiting for the reviewer's comments to come in. When we are revising the report, we will compare these methodologies and try to resolve as many differences as possible.

CHAIR APOSTOLAKIS: The question in my mind is, and I know you've answered a few times but it's not clear, probably because I don't understand this. It seems to me that the DFM guys can produce qualitative results that are useful without resorting to any probabilities or transition rates, and you can't. Now you say that you can, so that's something that I would like to see.

MR. ALDEMIR: You can see these -- you can regard each of these squares as a placeholder, non-zeroes as placeholders. You can regard them, if you want to make your life simple, we can regard them as ones, any time you have a non-zero probability, and that tells you how we can do that qualitatively. This is the --

CHAIR APOSTOLAKIS: Arabic.

MR. ALDEMIR: Well, hopefully these are all going to be Meccanite. Incidentally, what we are doing here --

MR. HICKEL: It's Greek, George. It's Greek.

CHAIR APOSTOLAKIS: If it looked Greek to me it would be okay.

MR. ALDEMIR: It's too small, and the resolution isn't that good, but these are lambdas and mus, which is Greek, yes. So eventually, the reason why we called it Markov is because of this, and this is a Markov process, and this has the properties of Markov. But as you will see in a little while, we can take this model, irrespective of the numbers we produce, and we can generate dynamic --

CHAIR APOSTOLAKIS: That's what I want to understand.

MR. ALDEMIR: Sure. Okay.

CHAIR APOSTOLAKIS: Now the last one that has a word that is very popular, "importance".

MR. ALDEMIR: This is importance defined after Lambert, but it is not one of the popular importance, but it's Lambert.

CHAIR APOSTOLAKIS: Who is that? Is that --

MR. ALDEMIR: Yes. This is from the paper published in 1989, so it's old. We don't use it any more, but --

CHAIR APOSTOLAKIS: Thesis.

MR. ALDEMIR: Pardon me?

CHAIR APOSTOLAKIS: That was his Ph.D.

thesis.

MR. ALDEMIR: No, no, no, no. Lambert was already working at that. Lawrence Livermore, I think.

CHAIR APOSTOLAKIS: Because it's typical of students, he published the paper ten years later, except for Mike here.

MR. ALDEMIR: This is, again, integration process. How do we do that? DFM I had already shown. Now coming to the point that interests you more, what we do is that we take the transition matrix, and we convert it into a dynamic event tree.

CHAIR APOSTOLAKIS: Who did that, the DETs?

MR. ALDEMIR: The Markov model, the transition matrix that --

CHAIR APOSTOLAKIS: I mean, who introduced the term? I remember somebody.

MR. ALDEMIR: Dynamic event tree?

CHAIR APOSTOLAKIS: Yes. Was it you?

MR. ALDEMIR: We did. I don't want to take undue credit, because I'm not too sure if it is Amandela and the associates, or us, but somebody -- we will use --

CHAIR APOSTOLAKIS: But Nathan Soo had something else.

MR. ALDEMIR: Yes.

CHAIR APOSTOLAKIS: What did he call it? DETM.

MR. ALDEMIR: Well, DETM is –– again, the word "dynamic" is there. Dynamic Event something - I forgot what the T stood for.

CHAIR APOSTOLAKIS: So the time has come for all these things to become useful?

MR. ALDEMIR: I would like to take this opportunity to point out to the foresight of Professor Apostolakis ––

CHAIR APOSTOLAKIS: When was the work trip you organized ––

MR. ALDEMIR: 1992. Maybe it's not the proper place, but I would like to acknowledge Professor Apostolakis' foresight. If he had not supported these activities through the Reliability Engineering and System Safety, none of this stuff would be here today. It would be very hard to publish. I remember I spent about a year to publish my first paper.

CHAIR APOSTOLAKIS: Flattery, but let's keep going now.

MR. ALDEMIR: No, I really am serious about it. It's not a flattery, but I am serious about this. Anyway, this is the -- we take the decision tree - sorry, transition matrix - represent it in a data structure of this sort, which corresponds to a dynamic event tree like you saw. This is showing you the actual data structure. This is on the left. It's showing how the event tree is going to look like from this data structure. Zeroes or Os stand for operational modes, Xs failed modes, plus means high, and I think -- no, plus means on and then X means off. So these are -- the symbols here are showing the state of the components, and how the system evolves. And this is overflow, overflow.

I'll skip through these. These are the algorithms that actually generate the trees.

CHAIR APOSTOLAKIS: Yes. Let's go to the real thing.

MR. ALDEMIR: Well, this is how the event tree looks like, basically, on the left.

CHAIR APOSTOLAKIS: That's it. I believe you. No, what I'm saying is there is no doubt that you have done your homework here. Take us to what really matters. So your --

MR. ALDEMIR: Once we produce the event trees - that we have done, pretty much - then the question is how you take this, and then we have the input structure that is compatible with SAPHIRE.

CHAIR APOSTOLAKIS: And we still have that problem how far to go, but as you said earlier, maybe it's --

MR. ALDEMIR: There is another --

CHAIR APOSTOLAKIS: Or something else happens.

MR. ALDEMIR: There are two issues here. One of them is, are we matching what is already in the fault tree through choice of initial condition, duration of the scenario, and so forth. That is one issue that can be resolved. The other part, how do we process after we input this time dependent information into the overall PRA, how do we process it, because right now none of these techniques will see the time dependence, including SAPHIRE, won't see the time -- they will immediately, the moment you start constructing fault trees, all that time information is lost. So we found a trick, so to speak, to process this, and DFM is doing the same thing. We are time stamping the events.

CHAIR APOSTOLAKIS: Why did you think it necessary to give us a history of SAPHIRE, but it was IRRAS.

MR. ALDEMIR: Completeness.

CHAIR APOSTOLAKIS: But I'm curious, several modules were written to compliment IRRAS. Is that correct?

MR. ALDEMIR: No, not compliment. That's a misspelling. Complement with an E, not I. This is -- at the beginning of the talk I said, we are using the graphical input mode for DFM to illustrate how DFM results can be incorporated into SAPHIRE. This is how we can -- we are using the Markov model to illustrate, still qualitatively only, no numbers - how we can use the textual mode of input to incorporate the event tree into SAPHIRE. And this is the actual file, this is actual SAPHIRE input. This is the event tree on the left in detail.

CHAIR APOSTOLAKIS: So, Steve, you said earlier that, if I understand correctly, SAPHIRE experts at Idaho will get involved at some point?

MR. ARNDT: Of course, since this is research, if this proves to be practical and useful, we will transition this to the people at Idaho. We're already working with Curtis and other people.

CHAIR APOSTOLAKIS: But maybe on the way of deciding whether it's practical, you should bring them in a little bit and have them look at this.

MR. ARNDT: Oh, absolutely. Absolutely.

And part of Tunc's team includes people who work with Curtis on internships, and other things, as well as - I'll take a 20-second digression. Because this is a both technically challenging and important issue, we're doing extensive peer reviews of this work, and Curtis, as it turns out, is one of the peer reviewers of this work, so we're keeping the SAPHIRE people in the loop in a number of different ways.

CHAIR APOSTOLAKIS: Okay.

MR. ALDEMIR: SAPHIRE people know exactly what's going on. In fact, some of the algorithms that were developed were developed within the scope of another project. But the reason I wanted to show this slide is to address the practicality issue. Suppose I'm a utility and I don't want to get involved with these fancy methodologies, how can I do it? Well, this is one way.

We are also trying to generate the Markov model –- how should I say - mechanize the Markov model for generation procedure. DFM is already fail user friendly, so once you generate the event trees, the rest here - this is exactly how we would enter them from a practical viewpoint. So it's not speculation, you can actually do it.

What comes out of the SAPHIRE is a fault tree structure like this. Now these time events, these events will have time information in them. It is conceivable that that time information is inconsistent, because SAPHIRE has no idea what's going on except just looking at these. Each time stamped event is another separate event, so you will need to process the outcome to remove the inconsistencies. And we do the same thing with DFM. This is exactly step-by-step instructions as to how you would do, a practitioner with SAPHIRE would be doing this, and we have done it. I have two students right now working with Curtis on these issues in Idaho.

So, again, I just indicated the steps to show that it is doable. I have another 20 minutes, maybe. Any questions on the methodologies? Can I just –- okay.

CHAIR APOSTOLAKIS: I think we raised them as we went along.

MR. ALDEMIR: Now the benchmark, when Steve Arndt was talking about the benchmark problem, he emphasized certain features of it, and some time ago, about a half a year ago we published a paper in PSA `05 as to what requirements a benchmark model should have that it is representative of the digital technology as it exists today, and as it relates to nuclear reactors. And it's a fairly busy slide. I'm not going to go through every item, but two distinct, two main items are that we classify systems as loosely controlled coupled systems, and tightly controlled coupled systems.

Loosely controlled coupled systems are the ones where the failure events may be statistically dependent due to the process, as I showed earlier, how the -- through the dynamics, or it can be through direct wire connections, or communication networking. So we defined a number of properties that the benchmark system should have to test the effectiveness of the methodology that is going to be used for digital system evaluations. And the benchmark problem satisfied most of the requirements. It is also a practical system. It is representative of the feedwater control systems you've been operating PWRs.

Some of the requirements that are less relevant to systems used in nuclear reactor protection systems are not represented by the benchmark system, and as Steve Arndt pointed those out, networking, for example, shared external resources. And two particular challenging feature of the benchmark system are that we have some of the fault tolerance capabilities requires consideration of system history, which is particularly challenging issue in reliability modeling. And as I said, system failure mode may depend on the exact timing of failure modes.

How do we meet the modeling requirements that we have listed in NUREG 69.01, and again, I'm not going to go through these, this graph. So just to show how they meet them, first of all, requirement one - neither methodology, it basically says that it should not be based on purely operating experience. In other words, you observe certain failures, you build a failure model that only duplicates those, but cannot really look into the future. You identify failures modes, the only failure modes that you have for the system are the ones that you observe for the overall system, system configurations that lead to failure.

CHAIR APOSTOLAKIS: But you should be able to go to actual occurrences and convince --

MR. ALDEMIR: That's right.

CHAIR APOSTOLAKIS: -- yourself that you could have found them.

MR. ALDEMIR: That's why I quoted the -- I showed the artifact generation. We have actually -- we do have an artifact which we can predict it's going to occur. And it did happen in real life, not for the exact same system, of course, but it shows the potential of the methodology that it can. So both methodologies can account for all the features of the benchmark system. This is requirement two. Both methodologies make valid and plausible assumptions.

CHAIR APOSTOLAKIS: That's where I need to be convinced.

MR. ALDEMIR: Well, okay. That's why I gave a little example here, a little footnote. For example, I'll read this - "For example, the assumption that the process dynamics can be represented through a Markov transition matrix or a decision table of DFM, have been validated through previous work, lots of publications on this."

CHAIR APOSTOLAKIS: Have been, what did you say, validated? Wow.

MR. ALDEMIR: Well, depends on how you define the word "validated". Demonstrated, better maybe. "Similarly, normal operation of the benchmark system and its assumed failure modes were based on operating PWRs, as well as other digital INC systems encountered in practice. Both methodologies can account for all the features of the benchmark system, so the valid and plausible assumptions --

CHAIR APOSTOLAKIS: I really think I need to see solid arguments of the validation of the lambda J. I really do.

MR. ALDEMIR: You're referring to the --

CHAIR APOSTOLAKIS: Transitions.

MR. ALDEMIR: Transitions.

CHAIR APOSTOLAKIS: Okay. Let's go on.

MR. ALDEMIR: Both methodologies can quantitatively represent dependencies between failure events accurately. And, again, assuming that the data are correct, the modeling procedure is doing that, and these are other types of failures that the models can account for, intermittent versus functional. Both methodologies yield information that is usable by, let's say, a conventional methodology.

CHAIR APOSTOLAKIS: So your prime implicants or cut sets have been compared to Mike's --

MR. ALDEMIR: That's what I said we are trying to do.

CHAIR APOSTOLAKIS: Oh, you're trying to do. Okay.

MR. ALDEMIR: That is something that we should be -- we can do this qualitatively. Well, we tried to resolve the --

CHAIR APOSTOLAKIS: No, I'm not talking

about the numbers. I'm talking about here is what they found.

MR. ALDEMIR: Right.

CHAIR APOSTOLAKIS: Eleven prime implicants that Mike mentioned. Here is what we found, and if we look at them, they're almost the same.

MR. ALDEMIR: Right. Well, we'll do that . We'll do that.

CHAIR APOSTOLAKIS: Okay.

MR. ALDEMIR: Okay. Also, they yield enough information, or they model the system in such sufficient detail and completion that the non-digital IC system portions of the scenario can be properly analyzed, and so we are not just concentrated on software issues, and that relates to the question raised earlier. Well, this is what we would observe in the analog systems, as well. True, but the combination may produce new results.

CHAIR APOSTOLAKIS: So you guys are taking now for granted that we are looking at the system centric approach, right? This is what you're doing, you're looking at the system itself, and the software is just embedded in it.

MR. ALDEMIR: That's exactly right.

That's the philosophy we have adopted in the beginning. And, again, as Steve --

CHAIR APOSTOLAKIS: But for actuation systems, that may not be what you want to do.

MR. ALDEMIR: Right. But this is something that, again, how are we going to implement --

CHAIR APOSTOLAKIS: I understand.

MR. ALDEMIR: This is a future issue, but maybe in a kind of hierarchical fashion, use the classical first, then use DFM, then you go to maybe more detailed Markov, or maybe put DFM in the probability mode.

CHAIR APOSTOLAKIS: Are there any plans to look at very simple actuation systems?

MR. ALDEMIR: Yes, I think they do. The second benchmark here we talk about those.

CHAIR APOSTOLAKIS: Okay.

MR. ALDEMIR: Now, challenges. They have substantially steeper learning curves and more labor intensive than conventional event tree/fault tree methodology, but they can be alleviated by developing user-friendly tools. And this is also in the further future plans, not near future.

The other challenge, this has come up

during this meeting through and through, is that the failure data used by either methodology for quantification not necessarily credible to a significant portion of technical community. However, as has also been pointed out, there are efforts to remedy this. And also, both methodologies can be used in a purely qualitative mode to obtain information about the important failure modes of the system, even the numbers are not relevant.

And, again, another requirement that we would like to have is that the methodologies don't require highly time dependent, continuous plant state information, and these methodologies do. Depending on what system we're talking about, if the physics are there, if the process is complicated, there will be no way around it. Otherwise, you are not representing your system. We've got to do this. If, on the other hand, the system is simple actuation system, you don't need fancy dynamics and fancy methodologies, or a lot of states.

We haven't even addressed in this problem the communication issues, for example, in these digital systems, for example, which may require a large number of states. But if they don't, simple actuation systems, maybe even the conventional method would work well. So in that respect, the hierarchical approach could probably be better, use the standard fault tree/event tree approach. You want to check your results, go to the DFM, maybe, and then either normal mode, probabilistic mode, or maybe go to a more refined model. So these are, again, speculations as to how we can practically implement and validate these methodologies against each other. So, in other words, kind of –- I don't know if validation is the right word, or verification, but basically, to make sure that the results that we are getting make sense.

And I think I'll just summarize and leave it to Steve to talk about future work. So we have basically specified a digital INC system that can be used to evaluate methodologies proposed for the reliability modeling of digital INC systems using a common set of hardware/software/firmware states. The benchmark system specification includes procedures for system component failure mode identification and failure data acquisition. By failure mode identification, I mean we are doing an FMEA, and that's in the report, as well.

We have used an example initiating event to illustrate how these methodologies, the dynamic flow-graph methodology and Markov methodology can be used for the reliability modeling of digital INC systems. We chose these methodologies because they were identified as the more promising methodologies by NUREG 69.01. And both methodologies can be used to obtain qualitative, as well as quantitative reliability information for digital systems.

We have discussed the possible challenges with the methodologies, most of which can be resolved. And, finally, and maybe very importantly, some properties of the benchmark system considered in this first, that it may not apply to all reactor protection and control systems. So if for digital INC systems which may have less complex interaction between the failure events, the conventional event tree/fault tree approach may be adequate for the reliability modeling of the system.

CHAIR APOSTOLAKIS: At the workshop in August, are you planning to present this to the industry?

MR. ARNDT: Let me answer your question, then talk a little bit about this issue. The workshop in August is primarily going to be discussing what needs to be, and what is appropriate for a regulatory guide in this area. Obviously, this idea of a graded approach to the kind of modeling that is necessary is one important part of that. It's not the only important part, but is one important part of that. And the philosophy, based on what we've learned so far, will be discussed. I don't know if that answers your question exactly or not.

CHAIR APOSTOLAKIS: How would the stakeholders understand better what you guys are doing here? You will give a draft of the NUREG out? No.

MR. ARNDT: Not at that point. We're going to go through a process to both explain our ideas, starting with the presentation this afternoon and in the discussion in August, and then finally, the draft Reg Guide that we sent out for public comment. At the same time, get input in terms of both what they consider to be practicable, as well as whether or not they have significant technical problems with our approach. So we'll lay out what we think is necessary in terms of acceptance criteria and modeling detail, and all the other issues that we talked about, as well as a structure and strategy for what the Reg Guide would look like.

CHAIR APOSTOLAKIS: When in August is this?

MR. ARNDT: We haven't defined the date, but we'll probably define that in the next week or so.

CHAIR APOSTOLAKIS: Okay.

MR. ARNDT: Okay. In terms of the modeling effort, the next steps, and we've talked about some of these and whether or not they should be incorporated in this document we're currently working on, or wait for the next document, we're going to be finishing the detailed reliability modeling of the full benchmark system, look at all the different prime implicants for all the different scenarios, same for the DFM and the conventional approach. We're going to do a qualitative comparison of the different modeling methodologies we've looked at. We're going to do a qualitative evaluation based on the data from field data, as well as the fault injection experiments. We're going to incorporate that into the selected PRA and look at not only can it be done, but how difficult is it in practice, and then we're going to do this again for a separate benchmark, which looks at the other end of the extreme.

The idea of defining two benchmarks is to get as many of the different characteristics as possible in the two different benchmarks. This is an important to safety but not safety system that is a control system that has a lot of dynamic interactions.

The other benchmark, which is not defined yet, but is the one that's going to be an actuation system, will be a simpler system with less dynamics, but probably higher redundancy and issues like that, because it'll be a RPS, so it'll have different characteristics. And from that information, we hope to be able to make judgments, both in terms of our own modeling capability and we will require in a regulated application.

That's what we're going to talk about in terms of the dynamic analysis. This afternoon we're going to talk about some of the failure issues, software failure analysis, software database, and a little bit of the traditional PRA. And then at the end of the afternoon, we'll have a short discussion of where we stand in terms of our philosophy right now for the Reg Guide, and then the industry wants to make some oral comments.

CHAIR APOSTOLAKIS: Any questions from the persons around the table? Members of the public, comments, questions?

MR. ENZINNA: If you don't mind.

CHAIR APOSTOLAKIS: I don't mind at all.

MR. ENZINNA: I'm Bob Enzinna. I work at AREVA in the PRA Department. I have some experience creating PRA models for large INC systems. One comment. On your slide 51, you've got a matrix there that fills the page. And I'm noting that this example you have is fairly simple compared to what we have in real plants. If you were to do that model on a system that I've been working on recently, you'd need a much bigger piece of paper. And I'm concerned about how this would scale up to a large application, and I implore you to test that thoroughly before you put this out there and recommend its use.

CHAIR APOSTOLAKIS: Is your approach available to the staff?

MR. ENZINNA: We can talk about that. I can't make any commitments for my company without talking to the people that own the systems, but certainly, we're open to that.

The second comment I'd like to make, I'm having trouble seeing how this dynamic stuff is going to fit into my PRA. Ninety percent of what I need to model, I think, in the PRA is the protection system, the stuff that happens post trip. Most of this dynamic stuff, the dynamic issues that you're talking about seem to be applicable to control systems, like the main feedwater you're talking about, stuff that systems that mostly are out of the picture once the reactor trip occurs. Most PRA practitioners wouldn't even attempt to model initiating event frequencies with both in a model unless absolutely necessary, because they're not good predictors. The best predictor for that is data from operating history, and I would propose that a reasonable approach for these systems is to use historical data, use a conservative value until we got some operating experience to quantify those frequencies. I can't see putting a detailed model like this in place to estimate initiating event frequencies. And main feedwater, the example you've chosen, you know, has some credit and some accident sequences after trip, but it's not the primary defense. It's a non-safety system. The thing we're relying on the most in accidents like you're talking about are EFW system, feed and bleed, things that are safety assured, and are going to be actuated by the operator, or by the protection system. Thank you.

CHAIR APOSTOLAKIS: Thank you. Anybody else?

MR. NGUYEN: Yes. My name is Thuy, and I'm a loaned employee to EPRI from EDF, Electricity de France. I have a question. The digital systems, of course, do fail, and the research program you presented aims at modeling and understanding the failures, but they also provide, I would say, nice features that help in making the electro mechanical equipment more reliable. Is this also part of your modeling efforts and representing digital systems in PRA?

MR. ARNDT: Yes. And there's two issues associated with that. One is actually modeling whatever system it is to the level of complexity necessary to include the features that are important. For example, some of the fault tolerant features, the redundant features and other systems that are specifically designed to increase the reliability of the systems.

The issue there is, of course, data, but also to some extent you trade the level of modeling complexity with the amount of credit you want to give to these systems that are specifically designed to improve the reliability. So from a regulatory standpoint, we have a bit of a challenge there, because if we wish to take credit for the very good, and in most cases very effective mechanisms that modern digital systems have to increase their reliability, fault tolerant systems, high quality components, redundancy, and things like that, we also have to find a mechanism by which to validate they're operating correctly, and that they're being modeled appropriately. So we are aware of that, we want to include those features in our modeling, but the challenge is by including those features in our modeling, it adds to the complexity of the modeling. So yes, we are aware of those issues, and are looking at that as part of our research.

To go back to the earlier gentleman's comments, we are aware that there is a large number of systems that will probably be able to be modeled at a less complicated level than what we're talking about here. The point of this work is to understand where those thresholds are, as well as understand what is acceptable associated with modeling of the more complex systems. The system we chose right here is relatively simple in terms of the size of the system. More complicated systems can be modularized and dealt with in that way, if necessary, based on their complexity, and what actions they take based on the process. And I'm sure we will have some more discussions about this at the end of the day.

CHAIR APOSTOLAKIS: Any other comments? Okay. Thank you very much, Steve and Tunc, and Michael and Carl. We'll recess until 1:00.

(Whereupon, the proceedings went off the record at 12:01:37 p.m. and went back on the record at 1:06:09 p.m.)

CHAIR APOSTOLAKIS: Okay. We're back. Steve, you want to introduce the subject?

MR. ARNDT: Yes. We're now going to have a series of presentations led by Todd Hilsmeier, who is working on some of the data issues, and also the traditional reliability modeling methods, and some of the folks from Brookhaven National Laboratory. And at the conclusion of that part of the discussion, I'll lead a short discussion of where we are on development of regulatory guidance. With that short introduce, I'm going to turn it over to Todd.

MR. HILSMEIER: Thank you, Steve. My name is Todd Hilsmeier from Office of Nuclear Regulatory Research, and Division of Assessment of Special Project. And today, Louis Chu from Brookhaven National Laboratory, Gerardo Martinez from Brookhaven, and myself will be presenting development of a probabilistic approach for modeling failures of digital systems using traditional PRA methods.

The presentation outline will include a background information review of the project plan that we presented last year at the ACRS Subcommittee

Meeting, then provide the status of the project, then we'll go into the meat of the presentation, which Louis Chu from Brookhaven National Lab will discuss development of the failure parameter database for hardware, and Gerardo Martinez and Louis Chu will review the software failure events induced by software faults.

Regarding background information, NRC has a very comprehensive digital system research plan, and part of that plan is to develop probabilistic failure models for digital systems that can be integrated into PRAs using dynamic and traditional PRA methods, as Steve Arndt pointed out earlier in the day. And the digital system PRA project, which is a project that we're working on, uses traditional PRA methods to develop probabilistic failure model for digital systems. And this chart was presented earlier today by Steve Arndt, and it shows the NRC's digital system risk program. And as you see, NRC is developing dynamic methods and traditional methods, and both methods feed into the development of the regulatory guidance.

And though we're working on these methods in parallel, we're also working together to develop the methods through exchange of information, through peer review of each other's products, and through meetings to make sure that we're on schedule and meeting each other's needs.

Matter of fact, Bill Kemper and Steve Arndt, they're, in my eyes, are our customer. And because this project is very challenging, it's all about team work. And tomorrow we have a technical meeting between the dynamic group and traditional methods group to discuss future steps of the project. And then on Thursday, the dynamics group and traditional group will be going to NASA to discuss exchange of digital system data between the organizations.

CHAIR APOSTOLAKIS: Which NASA are you visiting?

MR. HILSMEIER: The headquarters with Dr. Dezfuli and Mike Stamatelatos.

CHAIR APOSTOLAKIS: Stamatelatos.

MR. HILSMEIER: Yes. Thank you.

CHAIR APOSTOLAKIS: An easy name.

MR. HILSMEIER: So we're looking forward to that meeting. This should be useful for both projects.

The objective of the digital system PRA project is to develop probabilistic failure model for digital systems using traditional PRA methods. And also, the objective is to provide input into the reg guidance on PRA modeling of digital systems.

This slide shows a high level summary of the research plan using traditional PRA methods to develop probabilistic failure model for digital systems. And the detailed research plan, as I mentioned earlier, was presented at ACRS Subcommittee meeting last year, and tasks one and two involves seeing how other industries model and manage digital system reliability. And this task was completed and presented at last year's ACRS Subcommittee meeting.

Task three involves documentation of our results of our work, and that's ongoing. And task four involves developing a failure mode effect analysis, and dependency analysis for digital feedwater control system, which is our case study.

CHAIR APOSTOLAKIS: Why not a fault tree analysis?

MR. HILSMEIER: Excuse me?

CHAIR APOSTOLAKIS: That was proposed in the mid-80s, right, to use fault tree analysis to identify failure modes? Everybody keeps saying FMEA, and I'm wondering why they leave fault trees out.

MR. HILSMEIER: We will be doing the fault trees during the development of the hardware and software. The purpose of the FMEA is to learn and understand the digital system.

CHAIR APOSTOLAKIS: Well, fault tree --

MR. HILSMEIER: Right.

MR. MARTINEZ-GURIDI: Well, in my mind, also what happens, when you build a fault tree, you already know what failure modes of the system are there, and so you use the fault tree to combine them to reach the top event. But before you build the fault tree, you need to know how each component fails, and what is going to be the impact on the system. So I see FMEA as a preliminary step to the fault tree.

CHAIR APOSTOLAKIS: But you don't say fault tree at all.

MR. HILSMEIER: But the fault tree is actually a --

CHAIR APOSTOLAKIS: Put FMEA, fault trees, all these things help you understand the system.

MR. HILSMEIER: Correct. Then task five, six, and seven involves developing a probabilistic failure model for the hardware of the system, with task five involving development of the failure rate database for hardware. And Louis Chu will be discussing this task in detail. And then task six and seven involves developing and quantifying the probabilistic failure model for hardware using a fault tree analysis. And tasks eight and nine involve developing and quantifying a probabilistic failure model for software, realizing that software is system centric. With task 8A, reviewing system failure experience induced by software faults, which Gerardo Martinez and Louis Chu will be presenting in detail today. And task 8A is completed, but is currently being evaluated by NRC. The dynamics group is evaluating our work along with myself. And the rest of tasks eight and nine involve development of the software reliability model, including answering questions, are software failure rates meaningful, and developing a linkage between software and hardware, and quantifying the model.

Once we establish the linkage between software and hardware in task ten, we'll combine the two models. Then in task eleven, integrate the digital system probabilistic failure model into the PRA. And the next presentation will be discussing task five.

CHAIR APOSTOLAKIS: Is the EPRI report you're referring to the one we discussed at the last meeting?

MR. HILSMEIER: Yes, it was.

CHAIR APOSTOLAKIS: You are still developing a position?

MR. HILSMEIER: No.

CHAIR APOSTOLAKIS: It's a year now.

MR. HILSMEIER: Right. We're not still developing a position, but this plan shows everything that we've done. We no longer are studying this guide.

CHAIR APOSTOLAKIS: Oh, you're not.

MR. HILSMEIER: No.

CHAIR APOSTOLAKIS: So you have a position.

MR. HILSMEIER: Well, we have a position as far as how it's useful to us in the development of the traditional PRA method.

CHAIR APOSTOLAKIS: Are you expected to send the formal opinion to EPRI?

MR. HILSMEIER: Steve would have to answer that.

MR. ARNDT: The EPRI report was submitted for our review, and I don't want to go into the gory details, but it was determined we would not review it formally for SER at that time, from an agency standpoint. The task he's referring to is learning from what was proposed in that methodology. At a future time they may resubmit it, and we may decide to write an SER against it. We looked at it from how we can use it to help develop the traditional model.

CHAIR APOSTOLAKIS: So the first one -- we have two reports from BNL.

MR. HILSMEIER: Correct.

CHAIR APOSTOLAKIS: Which one are you presenting first?

MR. HILSMEIER: The first one would be development of the failure parameter database.

CHAIR APOSTOLAKIS: Neither one has a title.

MR. HILSMEIER: Excuse me?

CHAIR APOSTOLAKIS: Collection of Failure Data, or a Review of Software Induced Failures?

MR. HILSMEIER: Collection of Failure Data.

CHAIR APOSTOLAKIS: Okay.

MR. HILSMEIER: And the objective of this report is to develop failure parameter database for digital hardware based on currently available data for quantifying digital system reliability models. And the approach analysis will be presented by Louis Chu from Brookhaven National Lab.

MR. CHU: I'm presenting our work, developing hardware failure database for digital systems hardware. The outline will include our objectives, review of available failure rate database, some comments on hardware reliability protection methods, and then I'll talk about use of hierarchical Bayesian analysis to come up with generic estimates of component failure rates, some conclusions, what we've done and some proposed additional data collection.

The objective of this task is to develop a generic failure parameter database of digital components based on currently available data in support of developing reliability models, such as fault trees, Markov models of digital systems.

CHAIR APOSTOLAKIS: So what failure parameters are you talking about?

MR. CHU: Component failure rates. Hardware component failure rates.

CHAIR APOSTOLAKIS: Of the computer you mean? Hardware --

MR. CHU: Yes, like microprocessors, memories.

CHAIR APOSTOLAKIS: Okay. All right.

MR. CHU: Okay. The approach we use is review of available methods and database, and then we came up to the understanding there's not too much out there, and we tried to do what we can with the available data, and we performed this analysis using data extracted out of PRISM.

This viewgraph summarizes the review of failure rate databases. The existing nuclear databases do not contain digital component failure rates. For example, IEEE standard, SPAR database, the T-book, the ZEBD, the Swedish database, they don't contain digital component failure rates.

CHAIR APOSTOLAKIS: What is the definition of a database? I mean, the IEEE standard is really the judgment of the people they polled, and this is qualified to be called a database? I mean, you could say it's a general term, but when I hear database, I usually have in mind something that has real data in it.

MR. CHU: Yes. What we have in mind is something that was estimated based on real data.

CHAIR APOSTOLAKIS: So IEEE standard wouldn't qualify.

MR. CHU: I thought some of that would -- I mean, they don't have digital components, but I thought some of that was based on actual data.

CHAIR APOSTOLAKIS: It's really expert

opinion. Now the expert opinion may have been --

MR. CHU: Based on some kind of data.

CHAIR APOSTOLAKIS: May have included experience with actual failures. And SPAR, SPAR is out kid. Right? We're trying to help them. Anyway, I mean, I'm nitpicking now. AP600, what do these guys say?

MR. CHU: It has some high level, I would say crude model of digital systems, and it contains some, you know, I call it scatter data. If you look into their database, they probably have some estimated failure rate of a microprocessor, or maybe a particular circuit board. And if you look more carefully, you try to trace how the failure rates were estimated. Typically, you found it's based on say Westinghouse proprietary data. And it's scattered in the sense that it doesn't cover all the components that you can think of in a digital system. And if you look at papers, you can see some -- some papers collect some data in a particular study, the estimated failure rate of a programmable logic controller. But then our attempt is try to come up with something generic such that when you do a study, if you collect specific component failure rates of the system you are studying, you can possibly use that data to update this generic failure rate.

CHAIR APOSTOLAKIS: Is it correct to say that of all these databases you have there, it's really the LER database that gives you real data?

MR. CHU: LER and EPIX gives you nuclear data.

CHAIR APOSTOLAKIS: EPIX doesn't have much on digital INC. Right?

MR. CHU: Well, even LER, you know, it's required, you're required to have LER. It has some reporting criteria, you have to violate tech spec, or you –- therefore, certain failure may not get reported. And another difficulty with use of LER is that often you see some failure, but then you don't know how many of the same components are being used at a plant, and how long they've been operating.

CHAIR APOSTOLAKIS: But they are real data.

MR. CHU: Right. And while I call the hardware reliability prediction method that is the military handbook to Telcordia and PRISM, supposedly they developed their model based on actual data, too. But then they came up with empirical formula that you just apply. In case of PRISM, I know, because we looked into the raw data and we extracted the raw data to do our --

CHAIR APOSTOLAKIS: What does PRISM stand for? Do you remember?

MR. CHU: My understanding is it's not an abbreviation of anything. It's just a name they chose.

MR. MARTINEZ-GURIDI: PRISM is a system that was developed by the Reliability Analysis Center, and PRISM is actually software that contains the database developed by this organization, that you can query to get the information.

CHAIR APOSTOLAKIS: And this center is military?

MR. MARTINEZ-GURIDI: No, it's a company.

CHAIR APOSTOLAKIS: Oh.

MR. MARTINEZ-GURIDI: They are mainly funded by Department of Defense.

MR. CHU: So --

CHAIR APOSTOLAKIS: SINTEF?

MR. CHU: SINTEF is an organization. I have its name. Let me see.

CHAIR APOSTOLAKIS: Yes, I know. It's a Norwegian company, but where did they get their data from?

MR. CHU: We haven't looked into it yet.

It just came to us. They came up with a data handbook dated 2006, I believe, so that's another source of information to look into. And the claim is that they have data to support the Markov model described in the IEC standard.

A few things on reliability prediction method. They include Handbook 217, Telcordia and PRISM. The problem with this method is that they attempt to capture many causes variability explicitly, and such attempt is too ambitious. That is, they introduce all kinds of high factors to adjust the base failure rates, and they use empirical formula. My speculation is that some of the factors, high factor they estimated based on actual data, but then they extrapolate.

CHAIR APOSTOLAKIS: Do you know what kind of review these things get?

MR. CHU: I know there's a Professor York Maledon, provide quite critical --

CHAIR APOSTOLAKIS: Just a professor?

MR. CHU: Yes. He had written several papers criticizing the accuracy of this type of method.

CHAIR APOSTOLAKIS: So really, they have not been reviewed --

MR. CHU: And he's only looking at it at the level of the results. And I think what needs to be scrutinized is how those factors were derived.

CHAIR APOSTOLAKIS: Oh, sure.

MR. CHU: In principle, they have some kind of internal document that's not available to us. But in general, you could say we could ask for those bases studies that came up with it.

CHAIR APOSTOLAKIS: They're probably like the pro forma shaping factors in a reliability analysis. You do what you like.

MR. CHU: Chances are, say in one case they came up with an estimate, you know, military equivalent is a factor three better than commercial one. And three may be used whenever you need you have a situation, but how accurate is. This is my speculation. Also, it's kind of based on what I know about the current data that they have. I'm going to show you in a later viewgraph. So use of empirical formula is not that accurate.

But on the other hand, I guess there isn't much other method out there, or data out there. They essentially add the failure rates of components to get a failure rate of a circuit board. And when it comes to redundancy, then you have to model separately. So they calculate the failure rate of a circuit board, and treating it as a series system, a system consists of components in series. And then if you have two circuit board, two redundant circuit board, then you have to model separate using something like fault tree or Markov model. So one issue is the accuracy of the empirical formula. And certainly, they didn't look into the uncertainty associated with it. At one point, I asked what about uncertainty? They just said there's so many uncertainties, they cannot account for it.

CHAIR APOSTOLAKIS: So large that we don't care about it. Right? So you actually talked to people who are responsible for these databases. You just didn't --

MR. CHU: I went to a training session on the PRISM software, and used that opportunity to ask some questions.

CHAIR APOSTOLAKIS: Very good.

MR. CHU: In looking at those reliability prediction methods, you know, they are software tools that implement the method. They only help you to estimate component failure rates, but they don't give you raw data. PRISM is an exception. It turns out in this database, they included the raw data in the form of a number of failures, number of hours. So we extracted this kind of raw data and used it in our analysis. The problem with it is there's very large variation in the data that is from different sources, you get very different estimates.

This viewgraph shows the data we extracted for one component. I think this is the data for random access memory, and the table shows - the first column is quality, typical, it's commercial or military. Environment GB means ground-based, and GM means ground-mobile. And next two columns are the raw data, the number of failures, the number of hours. And the last column shows a point estimate. Basically, for those sources that have failure, I just do a simple division. In this case, 12 failures in this amount of time, and you get some point estimate. If you look at this last column, you can see the point estimate varies from probably .1 to 10 to the minus 3. There's a lot of ––

CHAIR APOSTOLAKIS: A million hours.

MR. CHU: Yes.

MR. HICKEL: You've got to add a six on to those. I just have a simple question. And you're obviously trying to collect data on electronic components, but the thing that is probably most needed by the Agency is the ability to extrapolate that to something that might appear in a digital INC system. To be able to know you can make that extrapolation, don't you also have to know that the mode in which that equipment was used, the way it was environmentally qualified, and run in a power plant environment with tech specs and daily shift checks and all that sort of stuff. How do you know that data from, I don't know, NASA launch facility is equivalent to a control system in a power plant? How do you make that equation?

MR. CHU: This is why we use the hierarchical Bayesian analysis, that is in this method, we account for the variability from different conditions, different source, like those factors that affect the failure rates.

MR. HICKEL: Right.

MR. CHU: The factors could be the quality, could be the operating environment, and this population variability distribution captures such variability. And then when you do a specific study, you may obtain some failure data. Then you further do a Bayesian updating to specialize the failure rates.

CHAIR APOSTOLAKIS: You will talk about that at some point?

MR. HICKEL: Because I'm just betting that somebody from NEI is going to come in and say well, that's very interesting, but that data doesn't reflect anything we're using. I'm just trying to understand how specific this is to a nuclear power plant INC system.

CHAIR APOSTOLAKIS: You will tell us how to do that later?

MR. CHU: Later we have some suggestions to do additional data work.

CHAIR APOSTOLAKIS: No, no, no, the Bayesian hierarchical thing, you're going to talk about that?

MR. CHU: Oh, yes. I have two viewgraphs explaining that.

CHAIR APOSTOLAKIS: Okay. So let's take one entry here, take the first one, number of failures - 12, 633 million hours?

MR. CHU: Yes, million hours.

CHAIR APOSTOLAKIS: Million hours. So this was commercial, and this is a particular system, so this is the experience of some organization? You didn't collect each one.

MR. CHU: We didn't. When we asked about the source of the data, the kind of information we got was something like this source of data is warranty repair data from the manufacturer. You don't know what the manufacturer is, just a few words description.

CHAIR APOSTOLAKIS: No, but who recorded the 12 failures in 633 million hours?

MR. CHU: Manufacturer --

CHAIR APOSTOLAKIS: Oh, the manufacturer.

MR. CHU: -- of that particular component.

CHAIR APOSTOLAKIS: And the manufacturers are different in the different --

MR. CHU: It's not identified; therefore, I don't know. It could well be different manufacturers.

CHAIR APOSTOLAKIS: So the variability we see in the last column, is this variability due to different manufacturers, due to different environments?

MR. CHU: Yes.

CHAIR APOSTOLAKIS: Yes, both?

MR. CHU: Everything.

CHAIR APOSTOLAKIS: Both. Oh.

MR. CHU: Yes. And, of course, you can argue maybe you should treat commercial equipment separate from military, but if you look at the data --

CHAIR APOSTOLAKIS: The commercial - no, they're almost the same, aren't they?

MR. CHU: It's hard to tell them apart. That's another thing. By just looking at this data, it's hard to say that military equipment are better. Therefore –- and if you group them separately, you may not have enough data to do the analysis. And supposedly, this is the kind of data that PRISM or the Reliability Analysis Center used in coming up with their –-

CHAIR APOSTOLAKIS: Did they have this for all the components of interest to us?

MR. CHU: We extracted all the data that we were able –- that's in the PRISM database.

CHAIR APOSTOLAKIS: No, but I mean, you were able to find information like this for all the components we're interested in?

MR. CHU: I'm not sure, but there were some 30 components as defined in the PRISM tool. They have raw data, so we just extract all of them. We haven't tried to develop our model of the digital system, so when we do that, we'll know. But these components tend to be at a lower level, as you will see. That's kind of what we hope to do, at least do it once, and try to do a detailed analysis, understand the design, and learn from it. And then see how we can possibly -- the method can be simplified, the model can be simplified.

CHAIR APOSTOLAKIS: Now what if, let's say again the first row, look at -- we don't know how many components you have. Right? We just know the total number of hours.

MR. CHU: Right.

CHAIR APOSTOLAKIS: Is it possible that the 12^{th} failure was due to a design error, and that error was not present in the other 11, of course, not also in the ones that operated successfully. So why then -- I mean, just because we have number of hours and number of failures, why are we jumping into a failure rate? How do you know that there is a rate? Maybe one or two of them had a design error and they failed immediately. Do you know that all these 12 were components that operated for a certain period, and then failed?

MR. CHU: No, we don't have that information.

CHAIR APOSTOLAKIS: You don't know.

MR. CHU: All we have is what's in these two columns.

CHAIR APOSTOLAKIS: Okay. So then I'm arguing that you're making a pretty serious assumption there, that there is such a thing as a failure rate, because some of them may have had a design flaw and they failed right away. It was not a matter of failure due to random causes, lambda, usually lambda. I think these failure rates are so prevalent here, and very few people are questioning whether they're appropriate. So if you don't know what kinds of failures these are, then it seems to me getting a failure rate is probably not such a great idea.

MR. CHU: Well, we just don't have that information. Let me explain a little bit more.

CHAIR APOSTOLAKIS: I understand that you don't have it.

MR. CHU: The total number of hours actually is the sum over certain reporting periods, different years, so we added them up.

CHAIR APOSTOLAKIS: Sure.

MR. CHU: So there is a little more detail, information --

CHAIR APOSTOLAKIS: Well, let's take pumps, okay? And I start with 10 pumps in my test. I start them, two of them fail right away. They don't work at all, and the other ten fail at some intervals.

Is it reasonable to take the total number of failures and total number of hours they operated, and divide them and get the failure rate? Is that representative of what happened? No, because two of them never worked.

MR. HILSMEIER: Would that be kind of just failed to start, for the two that never started?

CHAIR APOSTOLAKIS: That's right. And maybe they had a design flaw.

MR. HILSMEIER: Right.

CHAIR APOSTOLAKIS: So here, I don't know why we're jumping immediately to the principle of failure rate. We don't know. Fine, we don't know, but we are adding more information here which is not based on what the database is telling us. And the reason I'm saying that is because you, yourselves, later will tell us 36 percent of the errors were due to some requirements problem.

MR. CHU: Those are software failures. These are hardware failures.

CHAIR APOSTOLAKIS: Yes, these are hardware.

MR. HILSMEIER: One of the limitations of this data is it's not failure mode specific, so we kind of had -- which you're going to need for fault trees.

CHAIR APOSTOLAKIS: All I'm saying is that most people would look at this table and think it's natural to go to the last column, and I'm not saying that it's natural to do that, because you don't know how they failed. You don't have to assume the failure rate exists automatically. I mean, if there was a design flaw, there was a design flaw. And strictly speaking, they should be accounted for in their unavailability calculation. We just don't know. If it was a failure rate, and this would be a point estimate.

MR. HILSMEIER: That's a good comment. We'll look into that.

MR. HICKEL: Got to have the pedigree to know how to do the calculation.

CHAIR APOSTOLAKIS: Yes. I mean, just taking -- that's why it's important to have a model in your mind when you do the data investigation. And here without really saying so, you assume the model, the exponential failure distribution.

MR. CHU: I'll put it this way, that's the only data we were able to find. And I'm glad --

CHAIR APOSTOLAKIS: The only data you were able to find is in the first four columns. The fifth column you created.

MR. CHU: Right. It's just providing an indication of a point estimate. We're not using that for other purpose.

CHAIR APOSTOLAKIS: I understand, but do you understand what I'm saying?

MR. CHU: Yes.

CHAIR APOSTOLAKIS: Okay.

MR. ELKS: I believe I can add some clarification. Carl Elks, University of Virginia. I used the RAC PRISM database, as well. And when I talked to them about this table, I was concerned much about the same issues as like where did you get this data, is infant mortality rate factored into it or is it not? The answer that I got back from their experts was the infant mortality rate was factored out, so this was stuff that occurred later in time.

CHAIR APOSTOLAKIS: They actually operated for a --

MR. ELKS: Yes. Now that's off-the-record from one of their vendors. Okay.

CHAIR APOSTOLAKIS: If that's the case, then the failure rate estimate makes sense.

MR. CHU: So with that column, we performed Bayesian analysis to derive population variability curves shown in this figure.

CHAIR APOSTOLAKIS: This is a two-stage Bayesian, is that what it is?

MR. CHU: Yes, but we used what's called hierarchical Bayesian, and it's said to be a more general method. But the underlying model is the same, the difference - the way I see it is only in solving the problem, how you numerically solve the problem. Like the typical two-stage analysis, people just discretize distribution.

CHAIR APOSTOLAKIS: Yes.

MR. CHU: Hierarchical Bayesian used Monte Carlo simulation in solving it.

CHAIR APOSTOLAKIS: Yes, alpha and beta, the parameters of which distribution?

MR. CHU: Of the population variability.

CHAIR APOSTOLAKIS: I mean, have you assumed the form?

MR. CHU: Yes. We made different assumptions, such as uniform exponential, log normal.

CHAIR APOSTOLAKIS: If it's exponential, you have only one parameter. Right?

MR. CHU: Right. No, on the population variability curve we assume either log normal or gamma.

CHAIR APOSTOLAKIS: Okay.

MR. CHU: But on these parameters --

CHAIR APOSTOLAKIS: Yes, I understand.

MR. CHU: -- they are further distributed. So the underlying model is that we have data from different sources, different plants, or different manufacturer, and this curve is used to characterize that variability. Therefore, the data from different sources has failure rates that are samples from distribution. And with the data from different sources, we go through the statistical analysis to estimate this distribution.

CHAIR APOSTOLAKIS: So then the question then that Dr. Hickel asked earlier, this is the answer, that you have a broad curve that represents different manufacturers, different environments, and so on. But then there is another assumption there that the environment and manufacturer of your application in a nuclear plant is part of this ensemble.

MR. CHU: Right.

CHAIR APOSTOLAKIS: Which is another assumption, because I don't know if those guys have Appendix B. Okay? Or the equivalent, so our environment is probably better controlled, so maybe we are on the low side. Maybe.

MR. CHU: Hopefully, if you have some data, then you further analyze it.

CHAIR APOSTOLAKIS: Oh, yes. You start with hopefully, you could say anything you want. But this is a good idea, I mean, trying to get there, and then maybe you can modify the curve to allow for the fact that we have all these controls and so on.

MR. CHU: Yes.

CHAIR APOSTOLAKIS: That's a funny looking distribution there, Louis. A little more tilted to the left and it would be really a strange beast. In fact, we would be wrong if you did it that way. Almost vertical there, isn't it? Is it freehand or - can't be because it's smooth.

MR. CHU: I don't remember how we came up with this.

CHAIR APOSTOLAKIS: So what is Mu-I?

MR. CHU: Mu-I, it's just lambda times T. This is a notation within the --

CHAIR APOSTOLAKIS: Oh, T to the minus lambda T. Okay.

MR. CHU: Yes, this is just a notation within the win BUGS, or hierarchical Bayesian method. This method is kind of advocated in the NRC handbook on parameter estimation.

CHAIR APOSTOLAKIS: Parameter estimation, yes.

MR. CHU: And we used it, and we recognize there's still some problem with the guidance here.

CHAIR APOSTOLAKIS: There's no problem with the method. The problem is what we just discussed. I mean, the assumptions that go behind this, is my environment, are my components part of this ensemble that I get.

MR. CHU: Yes.

CHAIR APOSTOLAKIS: That's really the fundamental question.

MR. CHU: Yes.

CHAIR APOSTOLAKIS: Should I stress the distribution on the low side to account for those, and if I decide to do that, how am I going to do it so I can defend it. These are the real issues here, whether you –- I know what this method is. It's okay, theoretically it's okay. Who are the Brookhaven Science Associates, by the way? You?

MR. CHU: This is the company that manages Brookhaven Lab.

CHAIR APOSTOLAKIS: Okay.

MR. CHU: It's formed by people from the

universities, and BATEL Lab.

CHAIR APOSTOLAKIS: I thought it was a group within Brookhaven, but it's a hierarchical base. Right? It's higher.

MR. CHU: I've shown an example of the kind of data, and we extracted data for 30 components. And WinBUGS is the software that we used.

CHAIR APOSTOLAKIS: Who developed that?

MR. CHU: I'm sorry?

CHAIR APOSTOLAKIS: WinBUGS, who developed it?

MR. CHU: I think some people --

CHAIR APOSTOLAKIS: Oh, it's a commercial --

MR. CHU: Yes, it's available. You go to the website, sign up for it and you can download it. It's some British professor, probably.

CHAIR APOSTOLAKIS: Some who?

MR. CHU: British professor. I have some reference. I don't recall the --

CHAIR APOSTOLAKIS: He spells bayes with a lower a B?

MR. CHU: Okay. It solved the model by performing simulation. In our analysis of these data, we assumed failure rates were either log normal and gamma distribution --

CHAIR APOSTOLAKIS: You mean the failure rate distributions were log normal, not the failure rates.

MR. CHU: Right. The distributions, yes.

CHAIR APOSTOLAKIS: And the generic distributions.

MR. CHU: Yes. And further, the parameters of the distribution --

CHAIR APOSTOLAKIS: So let's look at the results. Yes, this is fine, I believe, we believe.

MR. CHU: The result is that because the data is very scattered, so --

CHAIR APOSTOLAKIS: Don't you have a curve somewhere? No? Okay.

MR. CHU: Some results, two viewgraphs of results. The problem appears to be the error factor is --

CHAIR APOSTOLAKIS: Wait, wait, wait. What you are showing here is the average curve, isn't it?

MR. CHU: Yes.

CHAIR APOSTOLAKIS: The average curve, so you have average overall values of alpha and beta?

MR. CHU: Right.

CHAIR APOSTOLAKIS: And this is the curve that you are showing us. Okay.

MR. CHU: Right.

MR. HICKEL: Okay. Can I -- this list of components here, this is from LER, PRISM, RAC?

MR. CHU: PRISM.

MR. HICKEL: PRISM only.

CHAIR APOSTOLAKIS: Yes. The kind of data he showed earlier. So what do we learn from this, Louis? I see some error factors that are pretty significant there, 173.

MR. CHU: Just too wide.

CHAIR APOSTOLAKIS: Oh, I don't know that it's too wide. I mean, maybe that's the reality. Right? I would say that the four point date is too narrow. What is the message from all this?

MR. CHU: There's very large variability among different -- the same type of component from different manufacturer or different sources.

CHAIR APOSTOLAKIS: But explain the largest error factor, I presume this is not normal, right?

MR. CHU: Yes.

CHAIR APOSTOLAKIS: Is 173, and on the left you say error. What does that mean?

MR. CHU: No.

CHAIR APOSTOLAKIS: Component is error?

MR. CHU: No, it should continue to error detection or error collections.

CHAIR APOSTOLAKIS: Oh. Oh.

MR. CHU: That's one component. As to the definition of component, there's uncertainty to what does that mean when it says error detections/collection.

CHAIR APOSTOLAKIS: Is that the component? I don't know.

MR. CHU: We tried to get some explanation to the component, but these names are strictly extracted from PRISM, and in our report we tried to give some explanation of what the component - what we think the component --

CHAIR APOSTOLAKIS: But since you took that course, is it possible to call somebody and find out? I mean, the others seem to be components, but this one I don't know.

MR. CHU: Yes, I think it's possible. Yes. This large variation, if you compare this to say what you see in AP600 or in some PRAs --

CHAIR APOSTOLAKIS: Is that million hours?

MR. CHU: Yes. Next table is the same. I want to back up a little. Let me see. Like to point out one problem with assuming gamma distribution. This is based on some recent work by Hover, Bunere, Cook, some of the people working on the PRA project, actually. They look into the two-stage Bayesian analysis, and they recognize the problem with --

CHAIR APOSTOLAKIS: Where are these people?

MR. CHU: Let me see. A few of them are currently with George Washington University, but I think they're originally from European countries working on - maybe German or --

CHAIR APOSTOLAKIS: What's that name again?

MR. CHU: Hover.

CHAIR APOSTOLAKIS: Oh, I know him, yes. Okay.

MR. CHU: So for gamma distribution, it can be shown analytically that the likelihood -- function becomes the likelihood of a common incident rate model when the parameters are large. That means, the likelihood is not bounded, it goes to -- it doesn't die as alpha beta goes to infinite. And it's improper, and it has no maximum, and is esoteric of the maximum along a ridge. Basically, is asked when you work with this kind of problem that you truncate, whenever you use computer to implement it, you truncate and you lose information. That would be --

CHAIR APOSTOLAKIS: If I use log normal, don't have any problem.

MR. CHU: Right.

CHAIR APOSTOLAKIS: Good.

MR. CHU: Right. Kind of I want to make a remark - we've done this kind of analysis so many years, and all of a sudden we recognize there's a problem, so there are still things to learn.

CHAIR APOSTOLAKIS: Well, the papers by Hover have been out also for a number of years, but the question is how many people have read them. But we're using log normal most of the time, so it's okay.

MR. CHU: Right.

CHAIR APOSTOLAKIS: Ahh, conclusions.

MR. CHU: We developed a process for estimating generic failure rates.

CHAIR APOSTOLAKIS: So you are saying then that the best we can do it to use PRISM. Is that what you're saying?

MR. CHU: That's the only place I guess in the raw data.

CHAIR APOSTOLAKIS: You didn't get

anything from LER?

MR. CHU: LER, that's the suggested additional work, you try to collect more information from the plant so that you find out how many of the same equipment are being used at the plants, or how long they've been operating.

CHAIR APOSTOLAKIS: Well, maybe instead of expecting to get information from LERs that will help you find failure rates, maybe you can get some idea as to how better our components are, and then devise a means of changing the low tail of the distribution you have developed from PRISM to account for nuclear environments. Maybe that would be a way to go, because I don't think these people have the same quality controls that we have. And probably the low tail of the distribution should be further to the left. I don't know. I mean, if you disagree, you disagree, but I think that's an issue here.

MR. HICKEL: That's a very good idea.

MR. CHU: We did look into some kind of regrouping of the data, but I find it hard because there isn't enough data to do this kind of analysis, when you do a --

MR. HICKEL: You know, I really had a problem with one of the conclusions, and this statement just kind of jogged it into my memory. Your report on page 21 said that when you searched the LER database for failures in digital INC systems, you only got 18 records?

MR. CHU: That was probably for a particular type of component. Maybe we searched for microprocessor.

MR. HICKEL: Right.

MR. CHU: I think. That's the case, we are –- I'm pretty sure that that's the case. Again, LER doesn't necessarily record all the failures.

MR. HICKEL: Right. I fully agree. As a matter of fact, I would say that most of the plants that have a device that includes the microprocessor would report in the LER the name of the system, not the fact that it was a microprocessor failure. They report that such and such system failed, and that would give you a low count. But the other thing is, I saw the word you searched. You mean you did an electronic search of the LER database?

MR. CHU: Yes.

MR. HICKEL: Well, you are aware that on the NRC LER website, they've got the optical imaging going back to 1984. I take it you didn't consider anything that was a paper record that's just been put on as a PDF.

MR. CHU: We did the search of the system being maintained by INEL.

MR. HICKEL: Right.

MR. CHU: And I think it does go back to like 1984. That's about right.

MR. HICKEL: It does, but you can't electronically search it, so when I saw the word that you searched for microprocessor, my immediate reaction was well, that's interesting. How do you search a PDF on a file like that? You can't.

MR. MARTINEZ-GURIDI: I believe that the LER search system can be searched electronically. You can specify a certain string of characters, and it --

MR. HICKEL: Yes, but many of the records going back that old, they're images, they're pictures.

MR. MARTINEZ-GURIDI: Not any more. I mean, that was the case a few years ago, but nowadays, they have the electronic version to `84 where you can search electronically.

MR. HICKEL: Okay. Because I was going to tell you, I personally had done a search of LERs looking for digital systems, and it happened to be in an area where I knew the names of the plants, I knew roughly when they had changed out, and when they did it. And I worked at CE a long time ago, about 20-30 years ago. I searched looking for information about their core protection calculators, and I got about 160 something LERs that all involved that system. There were failures all over the place, different kind of combinations and permutations of something in test, and a guy uploaded a new data set without knowing that one of the other channels was bypassed. All that stuff is there. There's MOX failures, there's CPU failures, all of those, and I think that that LER database contains failure experience that's a lot more relevant than what you might find if you're trying to find out what the Air Force is doing with a missile tracking computer or something like that.

The reason is, it has to do a little bit with pedigree, and I think George talked about, we talked about it a little bit. It's the mode that the equipment is bought, procured, installed, commissioned, tested, operated with tech specs, and people that have to do certain periodic tests. This is not commercial electronics like your laptop at home. It's a very different variety of stuff, and I think basically, I think there's a lot more in the LER data system than you're considering in this evaluation.

MR. CHU: We've only done some kind of trial search of the LER. We knew that we will not have information on how many of the same components are operating, how long they've been operating, so we knew we're not going to be able to use it to come up with some estimates, so what we searched LER was just some trial search, see what we can find. We didn't try to use that to do any kind of --

CHAIR APOSTOLAKIS: Do you plan to do this kind of more detailed search?

MR. CHU: That's what we're suggesting to do. The last viewgraph talk about it, but I recognize the difficulty. Searching LER is one thing, you have to somehow get information from the plant, that kind of information.

CHAIR APOSTOLAKIS: The last bullet, really, I mean did you agonize on it a lot before you put it there? This is a consensus view of the project, that better data should be collected? Yes, Louis, go on. Just say yes. Didn't you learn from Steve? Please identify yourself and speak into the microphone.

MR. STONE: I'm Jeff Stone from Constellation Energy. I work PRA. What I was questioning is you're focusing on operational failure rates, per hour failure rates. Are you going to address how we're going to quantify demand failure probabilities in this document?

MR. CHU: Not in this document, because all we have is those data from the PRISM tool. Like George pointed out, in some situations the failure could be demand type of failure, but we don't have that kind of data.

CHAIR APOSTOLAKIS: How important do you think that is?

MR. STONE: I think that's probably much more significant than the operational failure probabilities.

CHAIR APOSTOLAKIS: He's right.

MR. HICKEL: The issue is you've got some spike where there's a demand, that you need that equipment to work. And in that period, it had better be working in that interval, but that's -- if he's got the hourly failure rate, getting that wouldn't be that difficult.

CHAIR APOSTOLAKIS: Well, that's something for you guys to consider. I mean, it's okay that you haven't done it, but it's certainly something that deserves --

MR. STONE: Well, I mean, there are two parts to a demand failure probability. There's a part that it can fail per hour, or is there some shock failure probability when it's actually demanded. So just question that. Thank you.

MR. ELKS: Carl Elks, University of Virginia. Just one final comment I had. In my experience working with this PRISM database during the past couple of months, I've done a lot of CIRCA design of these safety critical systems in the past, and the components that are actually in the PRISM database are relatively old. I mean, these are the things that you would see ten years ago in a design, even longer. I mean, if you go back and look at that thing where you see latch counts, comparators and stuff, we don't use those any more, these FPGAs, and PLDs, and things of that nature. And I talked with the PRISM people about this, and I said when are you going to update your database so that we get more contemporary components, and they were going well, as soon as we get the data in. So I don't know if that was your experience or not, that trying to kind of look at it from the point of view of actually what's out in the field, and what's actually in the database, sometimes are not lined up correctly. And that's it.

MR. CHU: Well, I guess Reliability Analysis Center at least has some means of collecting data. We didn't even try, but that's what I kind of suggest you do in this last viewgraph. Try to collect data from the manufacturer of the equipment for nuclear plants, I listed some of the names that I'm aware of. And another thing to do is contact the plants so that we can --

CHAIR APOSTOLAKIS: It seems to me that both comments really you should add to your future activities. At least think about, these were both very useful comments.

MR. CHU: Yes.

CHAIR APOSTOLAKIS: Okay. That's it?

MR. CHU: Yes.

CHAIR APOSTOLAKIS: Now you have an interesting sentence here -- you want to say something?

MR. NGUYEN: Yes. My name is Thuy from EPRI EDF. In Europe there had been recently a new directive against the use of lead in soldering, and as a result, we had seen new failure modes, new hardware failure modes that due to the new alloys used to solder the electronic components. Have you heard of that? That the industry has called the whiskers issue. It's because you have very thin metallic whiskers growing from the solder of soldering pots that create short circuits between the legs of the circuits. And so for us, it's a new kind of hardware failure. And there also this notion of single event upsets, which are the fact that now the electronic circuits are so small, the engraving is so fine that you can have, for example, a stray neutron, a stray particle that can create a temporary error in the circuit, that when you restart the system, everything works correctly.

CHAIR APOSTOLAKIS: It's probably a higher order problem. Some useful input here.

MR. CHU: Yes, thank you for the input. We don't have -- we are not manufacturers, and we don't have easy access to the plants, so these are the limitations, that I suggest that we try to do something.

CHAIR APOSTOLAKIS: On page 28 you have a sentence that I found interesting. "Failure mode, specific failure rates are required in the Markov model. However, no such database exists." Now this morning we heard that you can get those. I don't give up, do I? You say "no such database exists."

MR. CHU: When I said that, I'm referring to the type of analysis that's done using the guidance of IEC standard, where you develop Markov models, you talk about fail safe, fail and safe, safe --

CHAIR APOSTOLAKIS: Well, that's what we had this morning, didn't we? There were two states, fail safe, and fail unsafe?

MR. CHU: Right. But how do you estimate --

CHAIR APOSTOLAKIS: And there were lambdas.

MR. CHU: How do you estimate the split, or how do you estimate the coverage?

CHAIR APOSTOLAKIS: Yes. That's my question, too.

MR. CHU: Right. That's the difficulty --

CHAIR APOSTOLAKIS: I really think you guys ought to talk to each other more often, because these are interesting comments coming from the same project. And we were told this morning that this will happen, so it's fine.

MR. CHU: Yes. I guess tomorrow we'll have a meeting.

CHAIR APOSTOLAKIS: You will talk tomorrow?

MR. CHU: Yes.

CHAIR APOSTOLAKIS: Okay, Louis. What's next? I see your name again. You name is Gerardo?

MR. MARTINEZ-GURIDI: That's right.

CHAIR APOSTOLAKIS: It's not Gerardo like you were introduced. It's Gerardo, right?

MR. MARTINEZ-GURIDI: That's right. I can use both.

CHAIR APOSTOLAKIS: Okay. So now we go to the second report, Review of Software Induced Failure Experience. Is that correct?

MR. MARTINEZ-GURIDI: That's correct.

CHAIR APOSTOLAKIS: Very interesting report, by the way. Now this is here, 30 slides, 31, geez. You need all of them, Gerardo?

MR. MARTINEZ-GURIDI: Yes, we'll go over it. Hi, my name is Gerardo Martinez. I work for Brookhaven National Lab. I will be presenting our review of software failures in different industries. The outline of the presentation is to present the general objectives of the project, our approach to reach these objectives. We also developed a preliminary model of software failures that we would like to have feedback from you. Then we'll present a review of the software-related failures at domestic nuclear power plants. At that point, Louis Chu will take over to talk about the review of events of software failures at other industries and foreign nuclear plants, the scheme for categorizing software failures, a detailed description of selected events. And as you know, a lot of this work was motivated by some ACRS comments, and we will try to address them. Also, discuss briefly some of the methods available for assessing the reliability of software, and we conclude with some conclusions.

The main objectives are to get a better understanding of software failures, to present an approach for collecting these kinds of failures, and to try to address ACRS' comments in light of insights doing this in achieving these two objectives.

In general, our approach was to search the LER search system.

CHAIR APOSTOLAKIS: By the way, you have to be a little careful. Some of these comments were not ACRS. They were not in a formal letter from the committee, so when you address the comments, you have to make the distinction. You understand what I'm saying? If there is a letter from the committee, signed by the chairman of the committee, that's the ACRS position. If you have at the end added comments by a member, that's the member's comments. You can't call them ACRS comments, because other members may disagree.

MR. MARTINEZ-GURIDI: All right.

CHAIR APOSTOLAKIS: I know this is new to you, but the record will have to be careful, I think.

MR. MARTINEZ-GURIDI: Okay. I suspected that, but thank you for the clarification.

CHAIR APOSTOLAKIS: Okay.

MR. MARTINEZ-GURIDI: We also did a search for events in other industries, and we developed the model I mentioned.

CHAIR APOSTOLAKIS: These other industries, everybody keeps saying we look at other industries and learned something. Have we ever learned anything from any other industry? We never learn anything.

MR. MARTINEZ-GURIDI: Well, one thing that --

CHAIR APOSTOLAKIS: Is that true? Did you learn anything besides they don't know?

MR. ARNDT: We learned that they have different approaches.

CHAIR APOSTOLAKIS: Yes.

MR. ARNDT: Frequently what we learn is that they've looked at things, and they decided it's too hard, and they're going back to simpler models. Frequently what we've learned, and we'll talk a little bit about this particular study, is that for detailed models you need detailed analysis. So we've learned some new things, but mostly we validated things.

MR. MARTINEZ-GURIDI: If I jump ahead of myself a little bit --

CHAIR APOSTOLAKIS: Please, do.

MR. MARTINEZ-GURIDI: Something that we'll learn from looking at failure events at other industries is that software failures can lead to really catastrophic outcomes.

CHAIR APOSTOLAKIS: Oh, yes. Sure. But again, you have to be careful about --

MR. MARTINEZ-GURIDI: And the kinds of failure modes that happen in other industries are totally applicable to the nuclear industry, as well, so in that sense --

CHAIR APOSTOLAKIS: That's a good point, Gerardo. That's a good point.

MR. MARTINEZ-GURIDI: Yes.

CHAIR APOSTOLAKIS: So let's go to the meat of this.

MR. MARTINEZ-GURIDI: Okay. We developed this preliminary model of software failures to understand better the causes of these failures, and to understand how they propagate in a complex system. The main objectives were to understand these failures, and to establish a basis for eventually developing a model to quantitatively assess the probability of software failure. And at the very top we classify the causes of internal and external, and I will go into that a little bit as we move on.

Software failure there can be propagated to the debate, to the devices controlled by the software directly, such as the valves, for example, as it was mentioned this morning, to the entire system in which the software is embedded, and to the overall plant, or overall complex system. The propagation of the failure will depend on several factors, such as the overall context, the overall state of the plant at the time of the software failure, and the tolerance to the software failure of the software, the devices, the system, and the plant.

CHAIR APOSTOLAKIS: And that's where, again, I believe the classification we have requested of applications would be very useful. One of the ACRS comments has been please develop a classification of various applications, actuation systems, feedback and control. Like you have some in passing in your report, real time digital, non-real time digital system, communication failure, so all this stuff that would be nice to have seen. Okay.

MR. MARTINEZ-GURIDI: Yes. Well, to mention something about that, that's a task that we don't currently have at the lab, as far as I know. So I am aware that is something is relevant to our project, and that --

CHAIR APOSTOLAKIS: I think it is, because you're classifying failures. It would be nice for us to know which particular systems are subjected to certain kinds of failures.

MR. MARTINEZ-GURIDI: Absolutely.

CHAIR APOSTOLAKIS: Okay.

MR. MARTINEZ-GURIDI: Okay. Something that I think is also very relevant is that the potential for dependent failures, common cause failures are also very -- is a relevant issue for software-driven systems because the redundant trains or channels of a system may use the same or similar software. In general, many times they use exactly the same software. And, therefore, if that is the case, then the failure of the software means that all the trains in that system will fail, failing the entire system. So if these dependent or common cause failure occurs, then it may cause a failure of all the devices, or the entire system. And this is something that has been observed both in the nuclear industry, as well as in other industries.

This is our overall model. What we have at the top is the development of the software, the stages in which software is developed, starting from the system engineering and modeling task, which you define what the software is going to be doing, and how it's going to interact with the surrounding system and the surrounding plant. Then you go to a phase of requirements analysis, in which you establish in a more formal way what the software is supposed to accomplish. Then you start in the design phase to turn those ideas into an architecture of the software. Then you move in to generate the actual code. Then once the code is generated, of course, these are very broad steps, and this is simplified model. This is certainly more involved. Then there is some testing of the software, and eventually it's brought into operation and maintenance, and that's --

CHAIR APOSTOLAKIS: Our regulatory review right now is really focused on the top five. Right?

MR. KEMPER: Yes, that's true.

CHAIR APOSTOLAKIS: And we are trying to

bring the lower part back to inform, or to expand the review. Right? We are really focusing a lot on the five boxes you have up there.

MR. KEMPER: As far as process for licensing review and licensing - oh, yes. Absolutely. Yes, the top five are the only areas that we can concentrate for a new application, obviously.

CHAIR APOSTOLAKIS: Right.

MR. KEMPER: Because all the rest of it is subsequent to that.

CHAIR APOSTOLAKIS: Yes.

MR. HICKEL: But when the equipment is in operation, isn't it true that that box, that next lowest level, O&M, isn't that historically where there have been most of the failures related to the software, and the constants, and all that?

MR. KEMPER: That's been my experience, yes.

CHAIR APOSTOLAKIS: But when we're licensing, we look at the top five.

MR. HICKEL: Yes, but you're all supposed to be looking in the license at the processes and controls that are going to be used once they get it in the field.

CHAIR APOSTOLAKIS: Yes.

MR. HICKEL: Because that's where there's less control, in those boxes on the top.

MR. KEMPER: Right. That's a configuration management plan or something along that --

CHAIR APOSTOLAKIS: Okay.

MR. MARTINEZ-GURIDI: So all these stages are usually known as the software life cycle, and it's often interesting to know, that you may already be aware, is that errors made at earlier stages in the development are just going to propagate into later stages, as you know, and compound with errors that may be made at subsequent stages. And once the software comes into operation and maintenance, there may be some faults there which may not necessarily be manifested, latent faults in the software, and that's what we call internal faults, or that's what we call internal causes. These eventually can be triggered and actually occur into a software failure, which is the next box down, the failure of the software, which would include the common cause failure, as I was mentioning before.

The failure of the software also can be due to external causes, which is the box on the right, which we categorize into four main types, which would be one human error, you know, somebody who operates the software in an incorrect way, failure of support systems, such as the hardware in which it runs, the power supplies, HVAC or any other support system that the software requires.

CHAIR APOSTOLAKIS: So is it correct to say that the dynamic methods we've heard this morning deal with the four vertical boxes, failure of software all the way down to maybe status of the complex system, but they don't deal with the external causes, at least in the present case.

MR. MARTINEZ-GURIDI: I would like them to answer.

MR. ARNDT: They don't explicitly deal with external causes. As related to what the operational profile is, the likelihood of having a input that is unexpected by design, it does look at that, in terms of --

CHAIR APOSTOLAKIS: But not human error.

MR. ARNDT: But not human error or things like that.

CHAIR APOSTOLAKIS: Whatever, high humidity.

MR. ARNDT: Right. That's not explicitly --

MR. ALDEMIR: Tunc Aldemir, Ohio State. We don't deal with external causes in the sense of human error, cyber security, external events, but supporting systems, there is interconnection between the system we are dealing with and the rest of the system. That's what happens when, for example, you hook it up with PRA, the whole PRA. So not intentionally, but partially covered.

CHAIR APOSTOLAKIS: Okay. Very good. Thank you.

MR. MARTINEZ-GURIDI: And then if we could move down in this diagram, what we tried to depict, again in a simplified way, is how a software failure is going to propagate with the possibility of creating a major accident. So from failure of the software that you could potentially have, a failure of the devices controlled by the software, then the failure of the entire system containing the software, and then that could propagate to have some impact on the plant. And then you could have some recovery. Of course, recovery can be applied at any of these stages of propagation. You can have recovery at the software level, you can have recovery at the device level, you can recovery at the system level, you can recovery at the plant level. And then if the recovery finally fails, then, of course, you may have an accident, otherwise will be avoided.

All of these propagation will also depend on the overall context of the plant, the overall state of the plant at which this happens. If the failure of the software happens to happen when there is some unavailability for equipment, then the propagation will be more likely, or more severe. And, of course, these boxes at the bottom is basically operating environment of the software.

So, to summarize, we see that the software - we proposed that the software can be analyzed in terms of these two main types of causes, internal causes resulting from the development of the software, and the external causes, which is the environment of the software. And also, the propagation depending on the overall context. And we also acknowledge that the specific context that is relevant for the software is the so-called error forcing context that has been proposed as a triggering mechanism for the failure of software.

CHAIR APOSTOLAKIS: I think the dynamic methods we talked about earlier, and the same, I think, idea applies. As I tried to explain what lambda might mean, it's really the occurrence of the error forcing context, which may trigger the manifestation of a design flaw some place, so it's time-related. Please.

MR. MARTINEZ-GURIDI: Okay. Now I will move on to the actual review of software failures at domestic plants. We did this review to identify and gain insights into the nature of these failures in terms of characteristics, such as the specific causes of failures, the associated error forcing context, and to identify any dependent failure, such as common cause failures.

Our approach was to identify these failures by using the licensee event report search system. We searched for basically the entire period available, which is from `84 to the end of last year. All plants, all modes of operation, and what we did was to search for the key word "software" in the abstract of the LER. This, of course, leads to somewhat incomplete set, because it's possible that we missed some LERs, but our objective was not to create a complete database, but just to get a sample of the most significant, hopefully, the most significant events that have happened in the industry.

The search was complemented with six additional events from NUREG CR 67.34, which is a new reg that this was specifically written to address, failures in requirement specification, and they identify some additional events. Some of the ones identified in that NUREG we already identified with LER, but there were six additional that we had not identified. And we were aware of an additional event, which was an interesting event, that we also added.

CHAIR APOSTOLAKIS: So why weren't these events in the database? I mean, you say you searched the LERs.

MR. MARTINEZ-GURIDI: Yes.

CHAIR APOSTOLAKIS: Yet six events are in the NUREG report, and also were aware of one. How come it's not in the database?

MR. MARTINEZ-GURIDI: You mean how come it was not identified?

CHAIR APOSTOLAKIS: I mean, the additional event that you guys were aware of. How comes it was not there?

MR. MARTINEZ-GURIDI: Well, it was in the LER search database, but because we only looked for the key word "software" in the asterisk --

CHAIR APOSTOLAKIS: Oh.

MR. MARTINEZ-GURIDI: So it is possible that there are some additional LERs that have the software -- maybe, for example, one possibility is that they didn't use the word software. The people who wrote the LER might have used computer code instead of the word "software".

CHAIR APOSTOLAKIS: Well, why didn't you use computer code as a key word?

MR. MARTINEZ-GURIDI: Well, the problem is that there are many possible words that can be used, so if we use all those we would end up with a very large number of LERs. And we didn't have the resources to go over those --

CHAIR APOSTOLAKIS: So on the one hand we complain we don't have sufficient data, and on the other hand you say -- that's okay. Keep going. Now you tell me when to stop for a break. You decide what is a logical place to do this.

MR. MARTINEZ-GURIDI: I think that will be when I finish this, before Louis takes over.

CHAIR APOSTOLAKIS: Is that within a reasonable amount of time? You're talking about five minutes or so?

MR. MARTINEZ-GURIDI: I can stop at any time, of course.

CHAIR APOSTOLAKIS: You can stop any time?

MR. MARTINEZ-GURIDI: Yes.

CHAIR APOSTOLAKIS: Okay. So it's up to me, then. Okay.

MR. MARTINEZ-GURIDI: Okay. Shall I continue?

CHAIR APOSTOLAKIS: Yes, please.

MR. MARTINEZ-GURIDI: So using this process, each LER that was identified using the search was reviewed individually. And those LERs that actually documented a software failure were selected in the database, so we ended up with 113 LERs that documented some sort of software failure. And these database we characterize these failure events in terms of basically some basics, such as the unit that was involved and so on, but more importantly, we provide a brief description of the software failure, its main causes, its consequences, the error forcing context and whether it was an independent failure.

Some means, as we learned, was that 71 different nuclear units have at least one event related to software failure during the period that we studied, so software failures have occurred in a significant number of units. And as a conclusion, we see that it's quite likely that any plant that uses software supported systems could experience a software failure.

Out of those 113 LERs, there were 17 that documented two units, so the software failure was applicable not to a single unit, but two units, so overall we found 130 software failures.

Then I searched the last 10 years of the software failures we identified, which is comprised of 45 LERs, to try to classify them in terms of what was the software failure mode, and the cause of the failure. And what we found was that in 69 percent of the cases, the software failed with a failure mode, it runs but it generates a run results which are not necessarily evident.

CHAIR APOSTOLAKIS: So this is the fail unsafe mode that we were talking about earlier?

MR. MARTINEZ-GURIDI: I would say this is certainly --

CHAIR APOSTOLAKIS: I mean, this is the -- the guy from Virginia, Carl. This is one minus your coverage.

MR. ELKS: Yes, this would have to be definitely --

CHAIR APOSTOLAKIS: Yes, one minus the coverage.

MR. ELKS: You have to put this in the system. Error detection mechanism didn't --

CHAIR APOSTOLAKIS: No, no. You have to come here. I'm sorry. Repeat everything you said since this morning.

MR. ELKS: Okay. (Laughing.) It won't take long. In the context of our definition of coverage, which we stated this morning, this would be an uncovered fault. Exactly.

CHAIR APOSTOLAKIS: That's a pretty high number, isn't it?

MR. ELKS: Yes, 31 out of 45 events. We don't know what the total operational time that these things, 20, 30, 40 years, maybe hundreds of years of operational time. Ten years, okay. So it's a fairly high number out of an event, I would say.

MR. MARTINEZ-GURIDI: Well, something that I think is very important to take into account is that these failures cover everything, both safety-related and non-safety-related systems. And possibly most of the failures occur --

CHAIR APOSTOLAKIS: Well, your classification is important.

MR. MARTINEZ-GURIDI: We'll be happy to take it up for you at Brookhaven. My impression is that most of the failures occur in non-safety-related systems, that may not even have any fault tolerant features, may not have coverage at all, or may have a very low level of coverage.

CHAIR APOSTOLAKIS: But, Gerardo, then I would expect you to put a couple of sentences to that effect in the report, because I don't see that anywhere. And all I see is 31 out of 45, and that's kind of --

MR. MARTINEZ-GURIDI: In the report it is mentioned that we believe that most of the failures are in non-safety-related systems.

CHAIR APOSTOLAKIS: But that's somewhere else. It's not where it should be.

MR. MARTINEZ-GURIDI: You mean --

CHAIR APOSTOLAKIS: I'm sure in a report of this size it's somewhere, but when I look at the heart of it, conclusion C.1, you're saying "69 percent had the failure mode runs with wrong results that are not evident", and there you don't say anything else. That's pretty scary. You should put these qualifiers there, because a lot of people look at the actual conclusions.

MR. MARTINEZ-GURIDI: Thank you for your comment.

CHAIR APOSTOLAKIS: You are very welcome. Okay.

MR. MARTINEZ-GURIDI: Well, another point is that we think it is maybe a reason for concern to have software that is running, we run this stuff sometimes for pretty long periods of time, and just generating incorrect results.

CHAIR APOSTOLAKIS: I'm sorry. You say that later. It is later in the report.

MR. MARTINEZ-GURIDI: Yes, it is there.

CHAIR APOSTOLAKIS: Okay. We're going to go to the causes of failure, the main cause was software requirements analysis with 16 hits, about 36 percent. As you may already know, the software fails to do its function because it was not designed to perform that function.

Another perhaps more surprising result is that operation and maintenance also had a pretty high percentage of failures with 27 percent, and these were events that were –– these were problems, issues introduced while the software was brought operational into the field, and then somebody somehow made some - perhaps with the best intention did some upgrade thinking that they were going to improve the system, and it turned out that perhaps they improved what they were trying to improve, but the software failed for other reason.

In many cases we were able to identify the error forcing context. However, in some cases, perhaps all again due to the fact that systems are non-safety related, the software didn't really perform its function from the start of its operational life. And it may remain hidden for a long time, perhaps several years. And also, what we saw from the operational experience is that the failure may be discovered by indirect means, such as somebody perhaps noticed some problem somewhere else, did some calculation, and in the process of troubleshooting, they found out that there was a problem, and eventually traced it down to software.

In a fairly large percentage also, about 26 percent, there was some type of dependent failure, including common cause failure. And additional 13 LERs potentially also involve dependent failures. We are not sure because we couldn't -- the LER didn't have enough information to find out whether that was actually -- 25 positively where there was actually a dependent failure. So it was clear that the potential of software failures to cause dependent failures is the most rated, and that since dependent failures can be a significant to risk, then software failures also have the potential to be a significant contributor.

I think I can stop at this time, if you think it's --

CHAIR APOSTOLAKIS: Thank you.

MR. MARTINEZ-GURIDI: Thank you.

CHAIR APOSTOLAKIS: We'll reconvene at 2:55.

(Whereupon, the proceedings went off the record at 2:39:45 p.m. and went back on the record at 2:59:36 p.m.)

CHAIR APOSTOLAKIS: Take your positions. Okay, Louis. Tell us what is going on here.

MR. CHU: Okay. I'll continue the presentation. I'll start with review of events in other industries and foreign nuclear power plants. Summarize how we search for events, internet search is the most important part of our method for identifying software-induced failures, and I provided some example websites containing descriptions of events, or references to details of the events. Just like other internet searches, they tend to -- one thing lead to another. You identify one -- you look up one event, and then at the same time, you find ten other events, so kind of the number of events you can find grows quickly. But you find from different sources there's significant overlap, also.

We used our judgment to pick certain events that we feel that are interesting, and we did some more detailed analysis. The aviation accident is an area where we did more thorough search; that is, the NTSB Aviation Accident Database was reviewed to identify software-related failures. We also looked at NASA website, which provide description of NASA missions, and some of the missions involve failures, and software failure was the cause.

In searching the internet, of course, we come across many news media, newspapers, magazines, and university websites. And information about the events, the level of detail varies a lot. In some cases, it could be two sentences in the form of an email, and then you search more for it, you cannot find anything. In some cases, there are more detailed official reports. These are basically how we search for events in other industries.

In terms of foreign nuclear experience, we basically make use of this NEA report that provides descriptions of some digital-related failures. COMPSIS is a database that's being developed, and currently my understanding is that they are still developing guidelines and database structures. From that international operating experience on digital systems will be collected.

CHAIR APOSTOLAKIS: Several years ago there was an international corporation that was established to look at common cause failures for hardware, which apparently did very well. Is there any thought to have something like this on digital software?

MR. ARNDT: The common cause database is sponsored by the same organization that is sponsoring the COMPSIS database program, so there is some interplay between the people who are working on both the data structures for COMPSIS, as well as the data associated with that. They're both OECD.

CHAIR APOSTOLAKIS: But we are participating in this COMPSIS.

MR. KEMPER: Yes, definitely. In fact, I'm filling in for the project manager, who just got promoted, right now. Went to a meeting just a couple of months ago in Korea, and we talked about this. And Louis is right, we're right in the middle of developing guidelines, coding guidelines and the user interface at this point, which will ultimately be available to everybody in the agency, hopefully, from a data acquisition point of view. But there's about 17 international regulators and research organizations participating in that right now.

CHAIR APOSTOLAKIS: Is the industry participating in any of this?

MR. KEMPER: Not at this time. We're still kind of kicking around ideas about participation and accessibility of the data. Right now, it's kind of protected, because a lot of -- some organizations across the world, they just don't want to share the failure data within their country, unless there's a reciprocity type of arrangement. But it's going to focus primarily on nuclear installed devices, that's the idea with COMPSIS.

MR. CHU: A little bit about screening of the events. Basically, in our search, we found a huge number of software-related failures, and we used judgment to pick some events that we think are interesting. Many of the events selected just based on their severity, the consequence of the failure. Some events were selected because they represent interesting failure modes, the failures associated with communication, or cyber security-related events. Some events were selected, such that we covered some specific industries.

In the end, we analyzed 48 events in 10 different industries. For each of these events, basically we tried to get detailed description of the event, and write up a description. And then we tried to categorize the failure modes of the software failures, and failure causes, failure consequences of these events, that as we develop, get a duration scheme for software failure mode and failure causes.

In addition, we tried to identify the sequence of events that trigger the software failure. In some cases, the precise sequence of events can be identified, in other cases it's just not clear, but it's obvious software error was involved.

I'll talk a little bit about how we categorize software failure events based on failure mode and failure causes. In general, it is hard to define, to narrow software failure modes, because failure modes may depend on the function of the software, and also depends on the level of detail at which you are talking about software failure. So in addition to reviewing software-induced events, we also did a literature review of software FMEAs, and see how other people define software failure modes, or if they do causes, and try to make sure the failure modes and failure causes that we have covers all those that others have identified.

Often in our review, we've often found

that the terms, the definition of failure causes, failure modes, and failure effect can be easily mixed up; that is, one failure cause may be the failure mode of some other study. A possible reason has to do with the level of detail. In a way, low level failure mode could be the trigger cause of a higher level.

By reviewing the events, and reviewing the literature, we came up with our way of categorizing the events. This table shows the high level failure modes we have defined. Essentially, we tried to define the modes in terms of the behavior of the software. And think of software could be a complicated system, consisting of elements, and then the elements can further be broken down into sub-elements, sub-elements can further be broken down, so based on that kind of thought.

MEMBER BONACA: I have a question regarding -- I mean, clearly, digital software in nuclear applications has specific requirements, and there are software requirements that are very specific in so far as verification, validation, and so on and so forth. To what levels do these kind of standards apply to the other databases that you looked at?

MR. CHU: I'm not sure I understand the question. Could you elaborate on that?

MEMBER BONACA: I'm saying in nuclear applications, software is subjected to specific requirements, which include verification, validation, testing, independent verification, a lot of steps to assure the quality of the software that's being implemented, and I'm just wondering about the other software that you looked at; are they subjected to similar requirements?

MR. CHU: We didn't specifically look into the specific requirement of other industries. I imagine there's a lot of variations in the industry, or in the military, aerospace, because more safety-critical systems are there. There might be more stringent requirement, but in our look, we didn't. We just looked at how failure occurred, and tried to categorize based on what happened.

MEMBER BONACA: Okay. So you don't have a sense of what the requirements may be. They may vary significantly from one application to another.

MR. CHU: Right.

MEMBER BONACA: All right.

MR. CHU: Okay. In this table at the high level, the left column, basically we call it system level failure mode. It's defined based on whether or not the software stopped running, and whether or not software failure occurred with a clear indication, so this relates to whether or not you can observe the failure, whether or not you're aware that failure occurred.

At the element level, we defined five software elements. They are kind of based on the function of the elements, input, output, communication, resource allocation, and processing. And for each of these elements, we have element-specific failure modes that are shown on the next viewgraph. And this viewgraph shows generic failure modes that are generically applicable to all the software elements.

This graph shows the element-specific failure modes. For example, communication failure mode could be failed interaction in sub-routine calls or in data communications. Resource allocation could be competing for resources, priority errors. Software failure causes, similarly we define software failure causes. For internal causes, we basically relate those causes to stages in the software life cycle. Essentially, faults were introduced and not detected during the development process, so they are due to errors in the development stages. And for each event, we tried to identify possible stages in the development of software where error was introduced. And these software faults are introduced during the development stages, and that is the quality of the software depends on how good a job you've done in developing it in each stage of the life cycle. Therefore, somehow, if we want to develop some quantitative software reliability model, we are going to make use of this kind of information, how good a job have you done in developing the software. So this kind of failure cause categorization can potentially help with that kind of work. This is just some high level failure causes. In our report, we have more detailed examples for each category of failure causes.

Some insights, review of software-induced failures in other industries. In general, events that took place in other industries, that ones that we analyzed in detail, tend to be more exciting, or have much more serious consequence, because you're getting events from a wider source from many other industries. And, in general, I would say the same type of failure could happen in the nuclear industry. Of course, keeping in mind that nuclear industry, the safety-related system, there might be better -- but in terms of say developing model, that kind of factor can be taken into consideration.

Some insights - incorrect implementation and omission of function are important failure modes. Error due to requirement analysis stage are the most important failure causes. The occurrence of error forcing context triggering a software failure is a reasonable way of considering software failures; that is, the software failure rate effectively is the rate at which the error forcing context occurred.

In some software failure events, we recognize that the failure occurs at the very low level. In one case, a bit stuck at one or zero trigger a sequence event causing a pretty serious accident. And so the implication is that in order to capture this kind of problem, you need to develop a pretty detailed level of model.

Some software failures involve softwares that are not application softwares. The operating system, the diagnostic software, communication software, so to capture this kind –- to identify this type of software faults or failures is quite difficult. And in quite a few instances we did find software common cause failures, the fact that identical hardware used identical software. Man/machine interface is a contributor to some of the events.

I have some description, a reasonably detailed description of four events, but they are pretty detailed. I hope that I don't need to explain them, every one in detail, because it's going to be pretty time consuming. But these four events all took place at nuclear power plants. The first three occurred in domestic plants, the fourth one occurs in Bill's Canadian plant. And they all involved software failures. For the three events at domestic plants, they all involve software associated with redundant equipment, like diesel generator sequencers, core power calculators, and regulating voltage regulating transformers. They all have identical hardware running identical software, so in principle, common cause failure could lead to failure of redundant equipment.

Maybe I'll try to explain each of these events quickly. Turkey Point diesel generator sequencer - it was during a test that they found that there's a software logic error, such that high pressure injection pump wouldn't start. This was discovered during a test. But my understanding is, before this was discovered, earlier there was another LER reporting pump failed to start event. And at that time, they couldn't tell what was the reason the pump failed to start. And when they recognized this problem, they went back and identified that this was the cause of that earlier event, so this is interesting.

Another thing is, again, my understanding is that it seemed to say it can happen only when you are testing, but if you look at that earlier event, it was actually a real signal. There is a real actuation signal, and the system failed, or the pump failed to start, so this issue might happen with reasonable and high likelihood. Of course, problem - you discover the problem and the bug is removed, and it's no longer a problem.

CHAIR APOSTOLAKIS: Let's go back. You say the error forcing context is the test?

MR. CHU: During test - okay, the error --

CHAIR APOSTOLAKIS: That's when they found it. But the first bullet under consequences says that even if it was a real event, you would not have responded properly to an SI signal, and units 3 and 4 were operating outside their design basis.

MR. MARTINEZ-GURIDI: What happens is the sequencer can operate in different operational modes, and there was some kind of switching where you can select which operational mode. And usually, it was selected to be in an automatic test mode, so in a way the sequencer was always in this automatic test mode. So should a real signal come, it will most likely find it in a test mode, and, therefore, it will fail to actuate. That's actually what happened in the previous LER that he was describing, that's exactly what happened. And they couldn't find out -- they didn't realize there was this connection of events. But then with the second event, they realized that every time the sequencer was in some kind of test operational mode, it will have this vulnerability, that it will not respond to a real signal.

MR. HICKEL: Was the fault unique to a software system, or was it unique to the function that was being implemented?

MR. MARTINEZ-GURIDI: Well, it was certainly a software problem.

MR. HICKEL: If I took the same function and implemented it using a bunch of AGOSTAT relays, if I could find them on eBay or something like that, I would not have this problem, it was unique to software?

MR. MARTINEZ-GURIDI: My understanding is that it was unique to software. The thing is that I cannot give you a positive answer, because this kind of detailed information, in most cases, was not in the LER itself, so we didn't know, have all the details to tell. But it was clearly stated that the problem was in the software.

MR. CHU: This is an example, we're limited to the information that's available in the LER. In some cases, you find some description of the event. They identify some failure, and then they said they sent the circuit board to the manufacturer for diagnosing it, and then we don't know what happened, so there are technical situation, too.

CHAIR APOSTOLAKIS: So, Gerardo, you say the problem was that the sequencer was continually on --

MR. MARTINEZ-GURIDI: On a test mode.

CHAIR APOSTOLAKIS: Test mode. And who did that?

MR. MARTINEZ-GURIDI: The plant decided to put it in that mode.

CHAIR APOSTOLAKIS: So is it because they did not understand what that meant, or it was just a slip? Because that's really, it seems to me, the error forcing context.

MR. HICKEL: That's right.

CHAIR APOSTOLAKIS: Right?

MR. HICKEL: Yes.

CHAIR APOSTOLAKIS: Not that the sequencer is executing the test, is that somebody put it in that automatic loop where it was self-testing all the time.

MR. MARTINEZ-GURIDI: But it was not an error. It's possible that the plant believed that put it in this operational mode was the safest way to have it, so it would be operational - continually being tested.

CHAIR APOSTOLAKIS: So the error forcing context then was not understanding what it meant to have it in that mode. That's the error forcing context.

MR. MARTINEZ-GURIDI: But, perhaps, that was the mode in which the sequencer should be.

CHAIR APOSTOLAKIS: Then there was a design error.

MR. HICKEL: I was going to say, it's hard to believe that somebody delivered a sequencer, and they didn't run a test to see that it sequenced the loads on the diesel at least once. So this has to be a mode where it was not the normal standby mode of operation.

CHAIR APOSTOLAKIS: But the reason why I'm bringing that up is because it's important to understand what the error forcing context is.

MR. MARTINEZ-GURIDI: Yes.

CHAIR APOSTOLAKIS: You really have to look for the context that creates this error, so either they didn't understand it, and that's the error, or there was a design error. I don't know. And if they were advised to do this, then whoever advised them did not have all the information as to the behavior of this. You have to look a little more deeply into what is the context within which the software does something wrong.

MR. CHU: The next event is an actual common cause failure that took place at Pilgrim. It involved loss of multiple vital AC buses. That happens during a storm, such that there is power transient, a voltage transient. Their regulating transformer was designed to regulate the input voltage within 20 percent of the nominal value, 480 volts. That is, if the voltage goes beyond that range, it just automatically tripped the transformer, and as a result, you would lose the vital AC bus. It happens during that event some of the voltage goes below 350, and indeed, that caused tripping of the transformer, and loss of multiple vital AC buses.

Core protection calculator problem at Palo Verde. This appears to be just a software was written not following the requirement specification; that is, the core protection calculators take analog inputs and compare it with some set point and determine if a trip is needed. The design is such that when two input modules are unavailable, core protection calculator should generate a trip signal, but it didn't. It was programmed to use the last known good value of the input, so it seemed to me, it's a simple error of not program following the requirement specification. This type of failure, of course, is a potential common cause failure, too. To trigger its failure, you have to lose the two analog channels, which is probably random, so it's not that likely you'll have redundant failures because of this software failure.

Ontario Hydro's refueling accident - this is an accident that involved quite a few independent events; that is, you have combination of four or five events that appear to be independent to trigger the failures. And as a result, there's a small loss of coolant accident. What happened was that the CANDU reactor can perform refueling while the reactor is on-line. They way it's done is that you have a fuel channel. You connect one fuel machine to one end, and another to the other end, and then you connect the fuel machine to the fuel channel such that it become part of pressure boundary, and you push from one end. You push the old fuel out, and the new fuel in, and then you reseal the ends.

During this accident, what happened is one fuel machine was clamped to the fuel channel, and something went wrong with the control, such that a spurious, some stimulate independent event triggered movement of the grade of the bridge, such that when it's clamped and you try to move it, it created a small LOCA. The combination of events that led to this involve, first, there is a software fault in the error handling software; that is, somehow the return address wasn't specified correctly. It was specified such that at the end of this error handling, it will go through the routine that will move the crane. And that's one event.

And then, first, you have to have an error on the computer, depend on trigger error handling such that the address will be pointing to the wrong place. And then this machine, this computer actually was not used to control the fuel machine that's already clamped. It's used to control some other things, but it was used to control this machine earlier, but still it was connected. The control is still connected to the fuel machine, such that when someone using this computer to control some other things, he generated an unrelated error, but it triggers the error handling routine, an error handling routine at the end transferred to the movement of the fuel machine.

Another independent event is there should be another protected computer there that should detect this kind of situation, and prevent it from occurring, but that computer was out-of-service at the time, so there are kind of four or five independent events.

CHAIR APOSTOLAKIS: Are they allowed to operate with this computer out-of-service? Was this a violation, in other words?

MR. CHU: I didn't see description of any violation.

MR. HICKEL: It probably had a procedure that said if the computer is out-of-service, you must manually do what the computer was going to do. That's typical.

MR. CHU: So these are some of the nuclear events. And then there are many other events in other industry. Some involve much serious accident. The blackout that took place two or three years ago has to do with some rates conditions. It was reported in one book written by a former CIA employee that CIA planted a virus in software that the Soviet Union bought, and it caused an explosion in a natural gas distribution system, and it was a huge explosion that the satellite actually detected the explosion. At the time, it was during the Cold War period. Initially, we were thinking maybe they are launching a missile. This is reported only in that book. It was discussed in some newspaper articles, but there was no official acknowledgment of the event. So kind of that's interesting.

And water treatment system at an Australian location, they have some computer control of their system, and the company, they hired a company to install the system. That company has an employee that for some reason left the company, but decided to cause some trouble, and he set up some wireless control of the water treatment plant, such that in 40 instances that he just opened the sewerage, such that it dumped sewerage into the river, or into a park. Eventually, he was caught when the police saw him doing something with a computer at the site boundary.

CHAIR APOSTOLAKIS: Again, I think these incidents would make much more sense within the classification system that classifies the applications.

MR. CHU: Yes. I guess, like one example about virus is the Davis-Besse event, where there's a virus that was introduced to the plant network, because they allowed some consultant access to the internet of the plant. So that's another virus-related event.

CHAIR APOSTOLAKIS: Okay, Louis.

MR. CHU: Let's move on.

CHAIR APOSTOLAKIS: What else? By the way, this classification of failure modes, on page C-33 of the Reliability Modeling Report, there is a classification scheme, which I'm not sure is consistent with what you are doing. So that's something you guys want to look into.

MR. CHU: Yes.

CHAIR APOSTOLAKIS: Okay. So where are you now, discussion of ACRS comments?

MR. CHU: Yes. This viewgraph, basically this task was carried out --

CHAIR APOSTOLAKIS: What was the comment? You are telling us what you did, but what was the comment?

MR. CHU: I guess it's a comment from one ACRS member.

CHAIR APOSTOLAKIS: No, no, no. What was the comment, not whose comment it was, what was the comment?

MR. CHU: One is looking at failure experience to identify --

CHAIR APOSTOLAKIS: Okay, yes.

MR. CHU: -- the failure mode frequencies. So we did this task in response to that comment. We developed a preliminary model of software failure, basically it give us high level picture, how we see software failure occurs. And we viewed operating experiences, and we developed a way of categorizing events. And regarding modeling of software failures, we feel it's reasonable to model it probabilistically, because the frequency is the same as the frequency of the triggering event. The question is how you estimate such frequency, but conceptually, I don't see a problem.

CHAIR APOSTOLAKIS: Are you talking about the fourth bullet now?

MR. CHU: Yes.

CHAIR APOSTOLAKIS: Well, I don't know how the statement of the constant failure is a reasonable assumption follows from what you've told us. Let's take the Turkey Point incident. I mean, I don't see where a failure rate could play a role there. The thing was useless, because it was constantly self-testing, so what is the failure rate? I mean, that was an error introduced from the beginning, and as you say in your slide, they were actually operating outside their design basis. I don't think that your statement there is supported by the evidence you have collected.

MR. CHU: The failure rate in that case would be the frequency that you have --

CHAIR APOSTOLAKIS: SI?

MR. CHU: Right. You have a demand.

CHAIR APOSTOLAKIS: No, because in a PRA, you would, under certain conditions, have the safety injection signal. Right? And then the next question is, what happens, is it executed correctly and so on, so you will need the probability there. The signal will come anyway, so the probability now is one that the sequencer will not respond correctly.

MR. CHU: Yes. It depends on where you start your calculation. There is a sequence of events that led to this SI signal.

CHAIR APOSTOLAKIS: Right.

MR. CHU: So the frequency of that sequence of events effectively is the frequency of this failure.

CHAIR APOSTOLAKIS: Yes, but this is because you know that the thing will not respond. But when I do the PRA, I'm doing a prospective analysis, so now the signal comes, and I know it has to be processed by software. What am I going to say? You're saying that in that particular case, it happened that the conditional probability was one, but that does not justify a constant failure rate.

I would say your first statement, the frequency of the EFC occurs, makes sense in some cases. In other words, the software operates, and then a set of conditions occurs, for which it was not designed, for example. Then the frequency of failure is the frequency of those conditions occurring. Right?

MR. CHU: Right.

CHAIR APOSTOLAKIS: It makes sense to have a rate there, but not in the Turkey Point case . It was useless. Any frequency that demanded operation from the sequencers was bound to –- I mean, would lead to a failure. There is a subtle difference, I think. Put yourself in the situation where you're actually trying to do a PRA, and now you have, in this new world, you have to consider the digital system as part of the system, the whole system, the response of the plant. Digital system is useless in this case.

MR. CHU: Right.

CHAIR APOSTOLAKIS: And it's not because of the context. The context is not something that applies to everything. I mean, based on what you have found, it seems to me that it's not something that is useful in general. In some instances, it is. Like, the classic example where airplane, the pilot tried to lift the landing gear when the plane was on the ground. I mean, there you can say yes, the software has nothing to do with this. It was used in a context for which it was not designed, although you might say the designer should have predicted that. Okay? So it depends on how you look at it. But in this case with the sequencer, it seems to me the context has nothing to do with anything. It was just an error.

MR. CHU: It is the sequencer event that led to the SI signal. But in case of PRA modeling, I agree that we need to look at, maybe instead of the model that in terms of probability.

CHAIR APOSTOLAKIS: Well, as we were discussing earlier, if the error forcing context was the misunderstanding of what the self-testing mode meant, then you might say the frequency of that misunderstanding is a rate, but I think we're stretching it a little bit.

MR. MARTINEZ-GURIDI: I think that something that is very important is that, as we discussed previously, there are some instances in which basically the software failure is already, is there all the time, basically since they installed the software.

CHAIR APOSTOLAKIS: Right.

MR. MARTINEZ-GURIDI: In that case, there's been no sense –- much sense in the failure rate. I believe that's what you mean to say. And the other case in which you have a software failure which is latent, and some error forcing context comes later, and then it triggers the thing.

CHAIR APOSTOLAKIS: Exactly. And I'm very pleased, actually, that we're having this discussion, because I think we're really getting to understand much better what is going on, and what we want to model. We have to be very careful what we mean by error forcing context, and what is the rate. So under certain conditions, I agree, there is a latent error, and under certain conditions it becomes real. Maybe the rate of occurrence of these conditions then makes sense to use, but in other cases, maybe it doesn't.

So that's something for future thinking.

MR. CHU: Yes, we have a next test to look at this kind of issue.

CHAIR APOSTOLAKIS: Yes, and that's great.

MR. CHU: Your comment certainly will be helpful. We'll try to account for all this.

MR. MARTINEZ-GURIDI: Yes, but I think the discussion also illustrates that it's sometimes, or many times it's very difficult to identify in advance when we try to do a PRA, what is going to be the error forcing context that are out there.

CHAIR APOSTOLAKIS: Absolutely.

MR. MARTINEZ-GURIDI: I mean, there are so many possibilities, that it's a humongously difficult thing --

CHAIR APOSTOLAKIS: You can talk to the HRA guys how they do it. In fact, tomorrow we'll discuss it. They start with a basic scenario, they consider deviations from the scenario, and then they ask themselves how likely are these things, they rely on expert opinion a lot. And I'm not saying you should do that, but that's one input to the process, because those guys have spent a lot of --

MR. MARTINEZ-GURIDI: I --

CHAIR APOSTOLAKIS: Of course, when you

deal with humans, it's a different situation. It's not --

MR. MARTINEZ-GURIDI: Yes, it appears to me that for software, it's even a more complicated issue, because software operates --

CHAIR APOSTOLAKIS: More complicated than human behavior? I don't know. I don't know.

MR. MARTINEZ-GURIDI: Because it operates at an even lower level. It takes inputs at the very lower level, it just takes data, so it's just a humongously difficult problem.

CHAIR APOSTOLAKIS: Anyway, I disagree with that second sentence in the fourth bullet. I think it needs more thinking, so let's go on to 27.

MR. CHU: Identification of error forcing context is difficult, in general.

CHAIR APOSTOLAKIS: It's difficult, sure.

MR. CHU: So there's always some faults remaining in the software. On the issue of system centric versus software centric viewpoints, system centric viewpoint includes interactions of the software with the rest of the plant. Conceptually, by considering the interaction, it is possible to identify many of the error forcing context. But a general issue still, I think, is difficult to, or is impossible to claim that one can find all the error forcing context, all the faults in the software.

CHAIR APOSTOLAKIS: But so what? I mean, that's why we have this research project. Right? I mean, if it was easy, it would have been done.

MR. CHU: Right.

CHAIR APOSTOLAKIS: The thing is, I don't understand your last bullet.

MR. CHU: Okay.

CHAIR APOSTOLAKIS: There is no contradiction. I mean, it's not a matter of contradiction, it's a matter of what makes sense to do. And go back to Turkey Point again, if I gave you just the software, and I told you this is the self-testing mode, you wouldn't find any problem with that. Right? You can't really say whether it's safe or unsafe, or what. It depends on where it is used. I mean, the software was doing what it was designed to do. And actually, I think the whole rest of the work that was presented today is really system centric, as I think it should be. Now there may be some instances, I mean, sometimes you use word and it freezes. I don't know whether that has to do with anything with another system, or with me, or whatever, maybe it's part of the -- but this is a limiting case, so I don't know that the word "contradiction" is the right one to use. It's what is useful and appropriate for us to do, and what we're dealing with is a nuclear power plant that's supposed to respond to certain emergencies in the right way, so that's the context within which we have to analyze these things.

MR. MARTINEZ-GURIDI: Yes. I think what we mean to say, what is exactly the meaning of software centric? I mean, if software centric means that we are only going to look at the software in isolation, then we are --

CHAIR APOSTOLAKIS: Yes, maybe as a separate component.

MR. MARTINEZ-GURIDI: Then we agree that that's not a proper way to approach it. However, what we see is that really software is never really treated in isolation, because --

CHAIR APOSTOLAKIS: In real life.

MR. MARTINEZ-GURIDI: In real life, because even when you design it, you are taking into account all this interaction, so you should take into account all these interactions with the plant.

CHAIR APOSTOLAKIS: Okay. So naturally, it should be system centric. That's what you're saying.

MR. MARTINEZ-GURIDI: If that definition includes that, yes.

CHAIR APOSTOLAKIS: Okay. That's what it is. You know, as you come to the fault tree the way we do it now, and then add an extra component, say digital system, you have to embed it in the fault tree and see how the components feed into it, they are commanded to do things. That's what -- it can't be just one additional component.

MR. CHU: Yes, I agree.

MR. NGUYEN: May I make a small comment, please?

CHAIR APOSTOLAKIS: Yes.

MR. NGUYEN: My name is Thuy, again. On this discussion of software centric viewpoints, there are a number of faults that we call intrinsic faults, that you can recognize as faults independently of the functionality of your system. For example, if you see a division by zero, or the use of uninitialized variables, or so on --

CHAIR APOSTOLAKIS: These are limiting cases that are not -- yes, sure. You should divide by zero. That's true.

MR. NGUYEN: Yes. But there are tools now that identify these type of faults automatically.

CHAIR APOSTOLAKIS: That's good. That's not my main concern. My main concern is, if I have a LOCA, am I going to mitigate it. That's really my concern. Now if you divide by zero someplace, then we're in trouble then.

MR. NGUYEN: Yes.

CHAIR APOSTOLAKIS: That's not my main concern. Okay?

MR. NGUYEN: Well, that's still a case.

CHAIR APOSTOLAKIS: How often do you divide by zero? I don't do that often.

MR. NGUYEN: Well, division by zero is only one --

CHAIR APOSTOLAKIS: I understand what you're saying. I mean, this is a limiting case, but that's not what should be our focus.

MR. NGUYEN: We made a number of analysis of safety software that has been in operation for quite a long time, and we did find --

CHAIR APOSTOLAKIS: But another argument I will make is that if you follow the system centric approach, eventually you will find these things. And we did that at MIT, a colleague of mine had designed control software for a mission that they were going to send to space and all that.

MR. NGUYEN: You may not have found it.

CHAIR APOSTOLAKIS: We found it using DFM, by trying to develop the decision tables, the student went there and he said oh, what is he doing here? He's dividing by zero. So it was found without really focusing just on the software, but trying to develop the –- but, anyway, your point is well-taken, but I don't think it's strong enough argument to abandon it.

MR. NGUYEN: No, no. It's just to say that there is no contradiction.

CHAIR APOSTOLAKIS: You can't talk to me from there. You have to come to the microphone.

MR. NGUYEN: It's just to say that the last bullet says there is no contradiction –-

CHAIR APOSTOLAKIS: I understand. Thank you. Are you done, Louis?

MR. CHU: Almost. Another ACRS comment was to look at software reliability methods, and review them critically, so we did some review, and in our report we documented –-

CHAIR APOSTOLAKIS: But it was not a critical review, because you say you will do a critical review later.

MR. CHU: Right. Our next task, we'll try to –- we'll get –-

CHAIR APOSTOLAKIS: You're going to come out and say this method --

MR. CHU: But I think all the foundation has been done.

CHAIR APOSTOLAKIS: You're going to come out and say this method is no good. Can you say that? Can we see those definitive statements at some point?

MR. CHU: We'll try to be more critical.

CHAIR APOSTOLAKIS: No, that's not what I asked. I didn't ask you to be more critical. I'm asking you to be truthful, because people usually are reluctant to say that, unless their own method is attacked, then everybody else is wrong, but that's different. I expect an objective assessment, Louis.

MR. CHU: Okay. We'll try. We'll try.

CHAIR APOSTOLAKIS: Formal methods, have you contacted the Canadians at all? I understand they have done something like this. Not exactly formal methods, but they borrowed from formal methods, and I don't know what they did, they formulated certain things using lesson learned from there, and they were very pleased with that. Ontario Hydro, have you talked to anybody there?

MR. CHU: No, no. We'll try to. It looks like formal method is a reasonable thing to try, even in terms of finding software faults. You use mathematical language to model your requirement specification, such that you can check. When you develop such a model, you think more systematically so it's not likely you'll make mistakes in specifying requirements, and the tools will automatically check for some kind of inconsistencies, completeness issues. And Nancy Levenson had done that in the Traffic Collision Avionic Systems successfully.

CHAIR APOSTOLAKIS: Well, SRI, I think, is doing -- SRI in California.

MR. ARNDT: George, the Germans and the Indians actually have also done work in this area.

CHAIR APOSTOLAKIS: Yes. It would be useful to see. Because eventually you may want to have a combination of approaches.

MR. ARNDT: Yes.

CHAIR APOSTOLAKIS: If this 36 percent of errors are due to requirements, you might say gee, my dynamic methodology doesn't quite fit that, but look what I do before I apply it. I do some formal thing to minimize it, I do something else, so the combination eventually probably will be -- they have different objectives.

MR. ARNDT: Yes. The big issue with

formal methods is that, at least as it's been applied in the nuclear industry so far, is that it's really more an error detection and error reduction methodology, as opposed to a modeling methodology. It's useful in other aspects of the digital research program plan, less so in the reliability part of it.

CHAIR APOSTOLAKIS: Yes, but if you tell me that I'm doing my reliability analysis using this method, assuming that I have already done these other things, then maybe that will give it a little more substance.

MR. ARNDT: That really gets to something that the U.S. industries also put forth as part of the EPRI methodology. The mechanisms by which you can, like formal methods, and redundancies, and fault tolerant techniques --

CHAIR APOSTOLAKIS: Okay.

MR. ARNDT: -- give you a higher likelihood that you're not going to have problems.

MR. CHU: And the method I think was recommended by the National Research Council, too.

CHAIR APOSTOLAKIS: Which method?

MR. CHU: The formal method.

CHAIR APOSTOLAKIS: As one of the methods that are available. Right?

MR. CHU: Right. Since we are trying to develop Markov type of model for digital system, and quantification of software failure rates or failure probability will be an important part of the model development. Currently, we're thinking about using Bayesian belief network method. Some European countries have tried it. It is a tool for performing quantitative analysis of decision making, and in our application, we will develop some kind of network, and one of the nodes will be say software failure probability, the quality of the software. And then we identify different things that affect the quality of the software, the failure rate, or failure probability of the software. And express the relationship in terms of some kind of conditional probability tables, and such tables certainly will have to be derived probably based on judgment, based on expert elicitation. In general, this seemed to be a reasonable way for quantifying software failure rates or probabilities.

Conclusion - software failures occur many different ways. Experiencing other industry is, in general, applicable to the nuclear industry. Some failure took place in such a way that implies very detailed modeling would be required. Some failures involve non-application software, that implies the type of software analysis needed to identify those problems. It's reasonable to model software failures in --

CHAIR APOSTOLAKIS: And that's where I am not sure that's correct.

MR. CHU: Yes.

CHAIR APOSTOLAKIS: And we need to investigate this idea of context and all that more carefully.

MR. CHU: Yes.

CHAIR APOSTOLAKIS: Remember, this is a subcommittee meeting that's supposed to be helpful. Right? I mean, it's not a final review of the project.

MR. CHU: We had a high level model for software failure. That part can be further developed, trying to look into this kind of issue.

CHAIR APOSTOLAKIS: Absolutely. Absolutely. Conclusion two.

MR. CHU: In terms of identifying software faults, it looks like there are many different methods. Each method, they have advantages and weaknesses. In general, you kind of want to use combination of them. But still in the end, most likely, you cannot assume there's no faults in the software.

CHAIR APOSTOLAKIS: The biggest problem here is not really finding faults, in the context of reliability, is what can you say about the probability of performance in the future, given that you have found faults, and you have fixed them?

MR. CHU: Right.

MR. HICKEL: The problem, George, is that I believe that there's -- just the data, I'd say the data right now shows that the rate of introduction of faults after its been turned over and is in use, is very high.

CHAIR APOSTOLAKIS: Yes, I agree.

MR. HICKEL: They include things like the vendor supplying the wrong set points, and that's not unique to digital, but it also includes all these -- there is a lot of experience about things getting changed in the field.

CHAIR APOSTOLAKIS: And the question is how do you model it?

MR. HICKEL: Probably your HRA is more associated with this then the digital software reliability.

CHAIR APOSTOLAKIS: We inject errors into the operators?

MR. HICKEL: No, they inject it into the equipment. Most of the time, the equipment catches it, and that's when you get an LER, thank God.

CHAIR APOSTOLAKIS: The common saying that you shouldn't fly an airplane right after its maintenance. Okay. I guess that's it.

MR. CHU: Yes. The things on the list we have has already been discussed.

CHAIR APOSTOLAKIS: Very good. Any comments for these gentlemen from anyone? Thank you very much. Very nice. And the next subject is the Regulatory Guide. I understand the presentation is not too long, but we are going to take a few minutes, so let's come back at 10 minutes after, unless the members disagree. You want 15 minutes?

(Whereupon, the proceedings went off the record at 4:02:39 p.m. and went back on the record at 4:16:55 p.m.)

CHAIR APOSTOLAKIS: Okay. Now we are talking about the Development of Regulatory Guidance. Mr. Arndt.

MR. ARNDT: Yes.

CHAIR APOSTOLAKIS: Have we seen this diagram before?

MR. ARNDT: Yes. I just wanted to mention a couple of things real quick before I go on. Two quick things, to fix it in the Committee's mind, because it's been an issue before. We're obviously going to be talking about this element here, the development of regulatory guidance, and this has inputs both from what our stakeholders were talking about, and what they're interested in, and the issues they have, but also the information we learned from the rest of the program.

Also, before we get out of here, I want to make a couple of quick comments to remind you who's doing what so you can get it straight in your head. The overall program plan, all the different areas, is being managed out of the INC Group, and I'm the overall Program Coordinator for that. The traditional methods that we talked about most recently, is being managed out of our PRA Group, Todd Hilsmeier is the NRC Program Manager for that part of it, and BNL is the prime contractor. The dynamic models, I also wear that hat as the Program Manager for that area. The prime is Ohio State University, Tunc Aldemir and his group, and he has a couple of subs, one looking at DFM modeling methodology at ASCA, and also the UVA that is working both on the development of actual interface with the system that we're working on, but also working on the modeling of the coverage space and things like that. So this is basically what the structure of the program is, so --

CHAIR APOSTOLAKIS: Are you getting any input from NRR?

MR. ARNDT: Yes. And as we move toward the regulatory guidance development, that involvement is going to expand.

Now as I pull this other one up, I want to also mention, we appreciate the opportunity to come and work with you. One of the things I just want to mention is at the last meeting, you really emphasized your desire to work with us, and work on intermediate results, so some of this has been watching sausage being made, to some extent. But we appreciate your comments and your review, and we hope to continue working with you in that area. And we can talk about that later after the end of the last presentation.

This is going to be some general ideas on what we think the structure and content of the regulatory guidance is going to be. As I mentioned earlier, this is a process by which we're trying to develop the ideas, get input, and work with the stakeholders before we send it out, the first draft out for public comment.

As we mentioned earlier, as part of the overall research program plan, we're developing the needed regulatory guidance to support risk-informed digital system reviews. To do that, we're taking the information that we're gaining from the other parts of this program, understanding the failure data, assessing the model, what models can be used, determining what systems need to be modeled at what level of detail, developing acceptable methods and acceptance criteria associated with that.

A little bit of reiteration. Industry has expressed interest in this area. We want to both develop regulatory guidance for regulatory applications of this method, but also to continually update the actual PRAs so they're consistent across the board, and model the digital systems.

MR. HICKEL: Steve, could I ask a question back on that last slide.

MR. ARNDT: Sure.

MR. HICKEL: You're saying as the licensees replace analog system with digital systems, their current PRAs are not keeping up with these changes. Now are you -- you're not expecting, or the staff, or NRR doesn't expect the licensees to modify their PRAs for non-safety-related control systems.

MR. ARNDT: We do not.

MR. HICKEL: You do not. Okay.

MR. ARNDT: And if you look at the way we implement risk-informed regulation, there's an evaluation as to whether or not the models that are being used for the particular risk-informed application are sufficient quality, completeness, and other things, to support that particular application. This simply is highlighting the fact that if you want to do something that happens to touch a system that happens to be a digital system, then you're going to have some challenges, if you haven't updated that piece, as well. If you don't need to do that, we don't need to evaluate it, and you don't need to have that application. But we're starting to see in a few very selected applications where that's starting to touch these kinds of issues.

MR. HICKEL: Okay. Examples being things like sequencers and --

MR. ARNDT: Examples being, for example, risk-informed tech specs. If you want to do risk-informed tech specs for various systems, and one of them happens to have control and protection systems, that's fine, so long as the modeling for that particular system is accurate to what's currently in the plant, and accurate to the level of detail that it models all the important aspects of the systems. If you want to exclude that particular system from your risk-informed tech spec, that's fine. But if you want to include it, then we need to establish some criteria as to what is a regulatorily acceptable digital system model for that application.

MR. HICKEL: Well, the main reason somebody might want to get relief is he's going to put in a system that's automatically tested to replace one that he used to have to go do surveillance on.

MR. ARNDT: That would be one example, yes.

MR. HICKEL: Okay.

MEMBER BONACA: A question I had, Steve, was a number of these replacements, I believe have occurred under 05.59.

MR. ARNDT: Correct.

MEMBER BONACA: And I would expect that industry will still try to use 50.59 to perform changes without having formal approval.

MR. ARNDT: There will be a number of situations where that will be the case, yes.

MEMBER BONACA: Okay. Now I'm wondering

about bullet number two, where I see that the industry has expressed interest in using risk-informed regulation, Regulatory Guide 1.174, as an alternate method for licensing the systems. And so I'm trying to understand --

MR. ARNDT: Some systems we have specifically stated we expect the licensees to bring them in for regulatory review.

MEMBER BONACA: Okay. There has been the clarification.

MR. ARNDT: Reg Guide 1.174 provides guidance on how to do risk-informed decision making. But as we've talked about, it doesn't provide specific criteria for digital systems. Now does it necessarily need to? Well, as we work this out, we'll find out what additional guidance, if any, is necessary. As you know, there's a series of guides to specific risk-informed applications, risk-informing the Q List, risk-informing the tech specs, et cetera. We believe the unique aspects of digital systems means you need some additional guidance.

Because of that, we want to look at issues associated with digital system modeling, as well as the other aspects of regulatory review that you need to do for risk-informed guidance; that is to say, how does the requirements in 174 for maintaining sufficient safety margin meeting the current regulations defense-in-depth philosophy, and performance measurement strategies apply when you do a digital system upgrade based on the risk-informed application.

This is basically a reiteration of what I've said a couple of times already today, our strategy for the development. Development and understanding of the characteristics, what are the things that might be necessary to model to have a sufficiently good model for these applications? Some of those were articulated in Reg Guide CFR 69.01 and various other work that's been published, and will be published. Is this a complete list, is it a list that has to be satisfied by every model? No. That goes back to the categorization issue that we've talked to, and I'll talk to a little bit later in this presentation.

Identify methodologies for modeling the systems. We've done that, and we're going to continue to do that. Develop an understanding of the data issues - that's a very large issue. Develop draft regulatory guidance or a draft regulatory approach - this is the guide that we're going to use. It's tentatively DG-1151, an approach to plant-specific risk-informed decision making for digital systems. We're going to have, as we mentioned earlier, a public meeting or a workshop to discuss our strategies for putting this together, and we hope to publish the comment - the draft for public comment in December of this year.

This is a very rough first guess at a structure for what the reg guide would include. There's a discussion of the modeling requirements, discussion of the issues associated with integration of digital system models into the full PRA model methodology, discussion of the data requirements. I expanded out and will highlight the uncertainty analysis issue here, primarily because 174 doesn't talk to it in great detail, and this is an area, as we discussed earlier, there's a lot of uncertainty associated with the data, with the models, with the context or operational profile that are going to assume that we want to have some explicit guidance associated with this.

The acceptance criteria - is the Delta CDF and Delta LERF appropriate, and if so, are additional guidance necessary? And then, how do you interpret the other issues that you need to look at for risk-informing performance measures, maintaining sufficient margin, defense-in-depth, diversity, those issues.

Here are some of the modeling requirements we are looking at, including - now to some extent this is motherhood. We want to model everything as best you can, but from these criteria, we want to focus in on what we care about when we are going to review one of these models. The model must account for important, relative features of the system under consideration. Model must make valid, plausible assumptions about the system characteristics, and justify these. Model must be able to quantitatively describe the dependencies between failure events, support systems, common mode failures, dynamic interactions, and if the model - if you choose not to model some of these things, demonstrate why they're not important. In very simple actuation systems, it probably is very easy to demonstrate why they're not important. In more complex systems, probably not.

Be able to differentiate between permanent and intermediate failures, distinction between multiple and single failures, issues associated with the complexity of the system. If the system is not very complex, then you discuss why it's not important, and why the model doesn't need to include it. If it is complex and you still choose not to model it, then we have a much more detailed requirement for understanding how you're going to deal with that. Understand the model must be able to provide the kinds of information that you need for inclusion in a PRA, cut sets, probability failure, uncertainty.

There's nothing to say that this can't be a multi-stage analysis, a stand-alone model that is then integrated with the PRA. But if you're going to do that, you've got to go back to how does that meet the criteria above for characteristics, and interfaces, and system dependencies, and things like that. Methodology must be able to incorporate the various accident sequences, and have enough detail so that if there's interactions with non-INC systems, that that's included.

Level of modeling detail - same kind of concepts; that is to say, not saying you have to use DFM, or you have to use Markov, or whatever, it's saying you have to use modeling detail sufficient to capture the important aspects of the digital system. The digital systems RNL issues, issue you brought up earlier, George, unique failure modes, if there are unique failure modes, unique characteristics of software failures and tests, some of the stuff that

Louis mentioned earlier.

If you want to look at simplified models, we would ask that you verify that the unique system characteristics that are not modeled in your simplified models aren't important. We want you to look at understanding how the data fits the model. If you data doesn't fit the model, or you're not capturing the unique characteristics of the potential failure modes in the data, we want to understand how you're doing that, and why you're doing it that way. Common mode failure issues, system interaction issues, and the last bullet there gets to the issue that we talked about earlier in the day - validate the events that have happened in historical record can be modeled by the level of abstraction that you have.

We hope to have some examples to illustrate what we really mean by these things. We'll probably inform that by our categorization issues that we've talked about today.

If it's an implicit integration, if you're going to do a fault tree/event tree-type model, this is less important. If you're going to do something more sophisticated, this is more important, in the same way that you would, say, do a seismic analysis, or some other kinds of analysis that is embedded in current generation PRAs. You need to include all the important interactions and dependencies, and include systems that would impact or would be impacted by the digital system changes.

Data requirements - this is going to be challenging for everybody, but we want to look at what data is being extracted, both in generic databases, the plant-specific or system-specific databases, particularly if we're going to use databases from vendors or parts manufacturers that may not be publicly available information, or may not have had public peer review, and what the limitations and biases, if any, are for those systems. Then look at if some of the data is being supported by test methodologies, be it reliability growth modeling for software, or some of the factor acceptance testing, site acceptance testing data, or specific data, specific testing methodologies to develop specific data like the fault injection methodology, understand what those are telling us, and how applicable they are to the particular delivered product, as well as how much of the system are they really covering.

In terms of review of the database, these are some of the issues we want to understand. The data collection hasn't been done in a systematic way.

Is it a good structure database, can we interrogate it, is there good configuration management for the measures, is the root cause analysis for the database entries appropriate.

One of the biggest challenges with LER database, for example, is you frequently only get very high level causes, the module failed. Modeling at the module level, and that is sufficient, that's great. If you're modeling at a lower level, or a higher level, you need to understand how that has been generated, so that's going to be an issue that we're going to look at.

Now some of this is the same kind of stuff that you would see in any PRA analysis. However, there are some unique aspects of digital systems, so we won't look at them in a unique way. We talked about model uncertainty earlier, look at model uncertainty, look at operational profile uncertainty, or context uncertainty, if you prefer, the knowledge of the possible input space, and the probability distributions associated with it, and data uncertainty.

Additional requirements - as I mentioned earlier, this is acceptance criteria explicitly laid out in Reg Guide 1.174. There may need to be some additional acceptance criteria for the digital systems. We need to look at how we meet the current regulations and defense-in-depth philosophy as embodied in 10 CFR 50.55 a(h), the various reg guides, 603, and the interpretation of how our regulatory structure currently exists.

One of the issues associated with risk-informed upgrade or risk-informed evaluations is a specific look at how the performance measurement strategies are going to be applied. In the case of a risk-informed digital system, that might include long-term validation of the data used, monitoring of industry-wide events to assure the assumptions continue to be valid. As the technology associated with digital systems changes, we want to make sure that the assumptions that was used in the digital reliability modeling also continue to be valid.

So, again, these are first thoughts of things that need to be included in a structure that would, I think, both give the NRC a relatively good assurance that the modeling is being done appropriately, at the same time giving sufficient flexibility to the industry to propose alternative methodologies.

The research into the current state-of-the-art methods is being used to help inform this regulatory guidance development, looking at a large number of potentially viable methods, developing acceptable methods. And as I just mentioned, we plan on making this a performance-based; that is to say, not prescriptive to a particular modeling methodology, but rather, defining acceptable characteristics of a modeling methodology.

The point of giving you some general ideas here is to see whether or not you seem to think this is a reasonable first approach for developing the guidance, and also to look at issues that the committee may think need to be included that we have not thought of at this point. Any comments along those lines would be much appreciated.

CHAIR APOSTOLAKIS: This is a pretty high level description, so it's hard to, at least for me, to come up with any substantive comments, unless my colleagues have something to say. Is the subcommittee going to review this guide as it is being developed, subcommittee meeting?

MR. ARNDT: The standard procedure, as you know, is once the draft is developed, it will be sent to the ACRS to either be reviewed before public comment, or waive review until after public comment.

You, of course, have the option to review it before it's sent out for public comment, if you choose.

Additionally, of course, as we go forward, we plan on having additional informational briefings to the subcommittee.

CHAIR APOSTOLAKIS: That's what I was asking. I mean, you do plan after you have some, let's say it's 40 percent complete, maybe have an information meeting and see what the reaction of the subcommittee would be?

MR. ARNDT: It depends on scheduling, and sequencing, but we could do that. Well, for example, we're going to have internal review of the rough draft, we're going to have the workshop that's going to talk about this in more detail because it'll be further along at that point. We'll get feedback from the stakeholders. At some point between then and the time we actually send it to the ACRS for review, we could have a subcommittee meeting to discuss that, among other things.

CHAIR APOSTOLAKIS: I think that would be advisable. So you think the next time we'll see this will be when it's really a draft of a regulatory guide, not before. Well, maybe -- if we have a subcommittee meeting to discuss other issues, maybe we can find a couple of hours to also discuss the --

MR. ARNDT: That would be very useful.

CHAIR APOSTOLAKIS: Yes.

MR. KEMPER: This is Bill Kemper. I think that's a good idea, George, because we wanted to try to discuss the software metrics project that just didn't work out for us, so we do want to get back with you in the next few months to talk about that, so maybe we can combine this at the same time.

CHAIR APOSTOLAKIS: That would be a great --

MR. KEMPER: I'm very much interested in getting all of your insight into this draft reg guide before we actually send it out for public comment.

CHAIR APOSTOLAKIS: Very good.

MR. KEMPER: Probably, I'm guessing, probably around October-ish time frame is what we'd be looking at from a calendar perspective.

MR. ARNDT: We'll work it out with the staff.

MR. GAERTNER: I'm John Gaertner from the Electric Power Research Institute. First of all, it's been a very interesting day. I really enjoyed learning these things, and the exciting things you have underway. And as you know, we, and our representation, the industry group, we support the risk-informing of this decision making for digital INC, and we support the use of the PRA. But a few things, Steve, that you said in this last talk leave me a little concerned, so I just wanted to point them out.

First of all, there seems to be a strong desire to incorporate the INC modeling deeply into the existing PRA as part of this effort, and I think that could be a mistake. It's appropriate, I think, to use the PRA to determine the acceptability of the digital INC from a risk perspective, but a lot of the assessments you're going to do are going to be bounding, and that'll be acceptable to show the safety of the INC system, but you don't want those bounding assumptions put back into your PRA permanently. And also, there'll be considerably uncertain, as we saw from the data analysis that we saw. And we have issues with aggregation - when we put things together in PRA, and some things are highly uncertain and some things aren't, or highly conservative and aren't, we don't like to aggregate them. So I think it may be in the best interest to keep the two separate, to a large extent, and not insist that the detailed modeling be incorporated into the PRA, necessarily. That's my one comment.

My second one has to do with defense-in-depth. I'm still concerned that it looks like we may be still expecting to have a high level of deterministic defense-in-depth, in addition to the risk-informed, and that would make some sense, even in Reg Guide 1.174, because where there's a lot of uncertainty in risk analysis, one asks for defense-in-depth. So I want to make sure that we're not just compounding, that we're not adding this risk-informed as an additional requirement on what we already have, so for that reason, I think we need to reconsider the current defense-in-depth requirements in light of the risk-informed approach that we're using. So I hope you'll do that in your reg guide. Thank you.

CHAIR APOSTOLAKIS: Okay. Thank you, Steve. The industry has requested time, Mr. Marion.

MR. MARION: Good afternoon. My name is Alex Marion. I'm Executive Director of Nuclear Operations and Engineering at NEI. And I do have a couple of comments I'd like to make relative to successful application of digital technology in today's nuclear plants, as well as in tomorrow's nuclear plants. But before I get into that, I would like to make a couple of comments about the last presentation from Steve on the reg guide. And I accept the fact that this is very preliminary thinking on the part of the staff, but this is extremely important. If it's not done properly, it will be a barrier to progress, and what I mean by that is, the regulatory process associated with applying digital technology will be so onerous that it will not be applied. And that's a disservice to just about everyone involved, including the NRC.

Based on what I heard today from the research activities, and it's all kind of interesting, it appears that the NRC is creating a situation where they're going to impose on the licensees through this regulatory guide to develop answers to some of the questions that were raised today. And these are questions that the NRC ostensibly is hoping to address through this research program, so we have to be sure as we go forward, if you take it to that level of detail in this document, that we understand, together understand what the expectations are, but more importantly, how to satisfy those expectations in a reasonable manner. And that's going to be the greatest challenge in this effort.

And to get back to John Gaertner's comment about risk-informing the process, we do support that, but we do want to make sure as we go through that process and document it in the regulatory guide and license amendments that will follow, hopefully, that it allows us to prioritize and identify those areas that are risk-significant that warrant attention. And I submit that everything we talked about today in terms of the research activities are not necessarily risk-significant.

We do want to engage the staff as we go forward, which includes the Office of Research and NRR, this is a very important activity for us, and we want to make sure it's successful. Within NEI, we agree that we need to make this as successful as we possibly can, and so the only way we can do it is work with the NRC hand-in-hand, identify the issues, prioritize them from the standpoint of risk, identify options on addressing those issues, et cetera, and moving the ball forward, if you will.

Timeliness of this is a concern on our part, especially with regard to new plant activities. Currently, the vendors are designing systems. We have systems that have been installed in other countries. There's an opportunity to start collecting data. I submit that in the presentation earlier this afternoon where four operating events were identified, it doesn't make sense, to me, that we worry about a 15 to

20 year experience with digital technology, given the pace of technology and its development. Okay? Just think about what's happened in computer science over the last five years. Okay? And the processes that we have in place at nuclear power plant, as you well know, is where there's an event where there's a problem, there's a root cause evaluation, and corrective action taken, so the relevance of these old events just doesn't seem to make sense to me.

Let's see. Conventional PRA methods, at this point, appear to be satisfactory if software, common cause failure, and fault tolerant design features are modeled in a conservative way. And we provided a document to the NRC that was developed by EPRI on defense-in-depth and diversity, and we're hoping that the review of that document can proceed in light of what we heard earlier today, and the comments on it. We need to establish some confidence in applying PRA technology, and I was pleased to hear that the research program includes benchmarking. That's extremely important. We think that is one of the key elements of making this entire process successful, because that gives us a reasonable time frame to start developing some data, and we support that.

And what I'd like to do is propose an integrated approach. We'll be thinking about, after we all debrief next week, we'll be thinking about sending in a letter to the NRC offering an integrated action plan of things that we think need to be addressed in order to make this process successful. There are analyses and designs that are currently ongoing for new plant construction. I know that Oconee withdrew their submittal for their upgrade, but I suspect that there are other utilities, well, I know there are other utilities seriously thinking about a submittal, so there are things that we need to identify, that we need to address now within the next six months. Otherwise, all of this activity is in jeopardy.

The draft reg guide and the August workshop schedules are extremely ambitious in light of what we heard today, but I still think there are some opportunities for addressing the low-hanging fruit, and get the process moving.

The industry would like to be a peer in the review of the research projects. It's kind of awkward to be sitting here at a discussion, where the committee members are commenting about a draft report that they have, but that report wasn't made publicly available. We could have offered some input and comments, and insights on that, as well. So at the appropriate time, we would respectfully request to be part of that peer review, because this is extremely important to the industry in a number of ways.

We are also interested in looking for opportunities for collaborative research. We have the NRC's research plan, we'll look at that, and hopefully in the not too distant future, schedule a meeting where we can talk about such opportunities and try to figure out how we can work together on answering those questions.

I mentioned the EPRI topical report that was submitted. I'd like to see that review progress. We did receive comments from NRR. Those comments, I think we can respond to. We generally agree with the basic thrust of those comments. I don't know if we should expect similar comments from the Office of Research. I don't know if the Office of Research was involved in putting those comments together or not. All right.

Over the long term, NUREG CR 69.01 was published, identify methods. There are a couple of things we want to say about that approach. As we go through evaluating digital systems and how to model them, we need to keep in mind a couple of things. One is, there are applications that deal with a specific threshold, digital applications under these conditions you open a valve. All right? Under these conditions you respond to a particular pressure reading on an instrument, et cetera, relatively straightforward and fundamental. Others are more dynamic with a feedback loop process, and we need to make sure that those two kinds of applications have to be dealt with in different manners. And I think you acknowledge that, at least based upon what I heard today. But the NUREG CR 69.01 doesn't differentiate between those two forms of applications, or two types of applications.

We've looked at all the experience with digital systems, specifically some of the software issues, or the software-related experiences, and we characterize a great majority of them as being basic configuration management. Make sure that the application meets the intended service it's going to see in the field, et cetera, and you make sure it's compatible with the design features of the system that you're applying it to, et cetera. That's configuration management, straightforward.

As we go through this process, we'll consider whether or not any specific guidance or encouragement is needed from NEI in reinforcing that message, but that seems to be extremely fundamental that we need to agree on, and I think ultimately the staff will agree on that, as well.

We need to differentiate, as you go through these evaluations of software failures, it would be helpful if you could differentiate between operating system failures and application failures. That's extremely important. I mentioned the point about relevance of aged experience. One other thing, and the committee knows from presentations I've made before, that I really focus on the process. If we can understand the process, we know how to get from Point A to Point B.

We want to be careful that we don't use, or we don't set up an environment or situation where the license amendment process by utilities wanting to submit these applications for NRC review, becomes the way that the NRC regulates digital applications in the future. And I don't mean that in a negative, critical manner. What's important, I think, and the way to avoid getting into that trap is to focus on the risk-informed decision making associated with these applications, and I think that that ought to be the first principle that we all agree on. All right?

We've had experiences with risk-informed applications that have been successful, and let's see if we can translate that, or transfer that to applications in digital technology, and that's where I think it's fundamentally important to stay focused so we don't lose sight of that.

That completes my comments. I'll be more than happy to answer any questions. Some of our industry team is here. I don't know if they want to add any additional comments, or any clarifications.

CHAIR APOSTOLAKIS: I was thinking about it also today, not only today, and I'm glad you mentioned that you would be willing to have some sort of collaborative research going on with the NRC. And, of course, as we all know, the fire modeling effort was a very successful effort. In the past, we've had common cause failure, common project, joint project. I think it will be very, very useful to try to do that. I think we have to be a little careful about the timing of it, so that the industry and the staff will have maybe some ideas that will evolve and then come together. But I would be all in favor for that, because I think this is a way to develop something that's practical, stakeholder views come into the picture early, and I can't think of any downside, really. So I, personally, would be very supportive, but I think the committee would be also very supportive based on what we have seen so far, so I would encourage you to pursue this. And I don't know now when it would be an appropriate -- and I also think the suggestion from Mr. Marion of having industry reviewers of these documents is not a bad idea. I mean, I don't know what the law says about issuing draft reports before they are draft, and so on, but if you can accommodate that, it seems to me, Steve, you're going to benefit a lot. And, again, it will be in the same spirit we're having these subcommittee meetings; you are getting input early in the process so you have a chance to respond, or at least you know what's coming down.

MR. HICKEL: It would seem they're members of the public, also, NEI.

CHAIR APOSTOLAKIS: No, but if you treat them as members of the public, then you have to wait until the time comes for members of the public to see -- I'm talking about the peer review that's happening now.

MR. MARION: We've been involved in peer review of other documents, and so the precedent has been set, so I'm just offering that we're still available to help out.

CHAIR APOSTOLAKIS: I think that --

MR. KEMPER: Yes, if I could just add my two cents. It's certainly a priority and a goal of the Office of Research to collaborate with industry whenever possible, and so I welcome that.

CHAIR APOSTOLAKIS: So it seems to me there is no --

MR. KEMPER: It's just a matter of us getting together and working out the details, the logistics. All right?

CHAIR APOSTOLAKIS: Okay. Good.

MR. KEMPER: Peer review, also timing is perfect for that, because that's also another initiative by our office, is to assure quality of our documents to get as good a peer review as we can, so if we could maybe work out some protocol here about who would be the person, as opposed to sending it out to the entire industry. I don't know if that would be the best solution or not, so we can work that.

CHAIR APOSTOLAKIS: You can work out these things.

MEMBER BONACA: I'm disappointed to hear about Oconee withdrawing the application.

MR. MARION: Yes.

MEMBER BONACA: I didn't know that.

MR. MARION: Yes, just a decision they made about two weeks ago or so. I don't know. Tony Harris probably knows -- obviously, knows more about it than I. Are they going to reconsider submitting it, or can someone --

MR. HARRIS: No. This is Tony Harris with NEI. I was at the last meeting with the staff, and I think, Bill, you were there, too. Duke was contemplating at that time whether or not they would withdraw. I know they are -- I can't fully speak for them. I do know they are working out the plan under which they would resubmit the application, but they have sent in a withdrawal letter.

MEMBER BONACA: I think to have on the table an application, it will be very useful, I think, for progress, I mean, on this plan, because it'll be ideas, and the perspectives I think that, hopefully there is -- somebody else will do that.

MR. HICKEL: Mario, or George and Mario, there have been a number of people that were contemplating digital upgrades to protection and ESFAS logic, and there were announcements I think that jobs were sold. And then subsequently they seemed to have gotten off track. Is there any input from NEI, is this being caused by lack of guidance, or what is the cause that these things are kind of falling by the wayside? Is it complexity?

MR. HARRIS: This is Tony Harris with NEI, again. We did meet with the NRC staff. We had an EPRI/NEI co-sponsored workshop in March, and we started looking through it, because you're exactly right; there are a lot of folks. And the concern with the industry is the length of time on some reviews - now whether it's caused by issues on our end in terms of quality, or some of the issues that you see in terms of unresolved technical issues, some of these things that take a long time. The process itself does take a long time, and it may be that it will take some period of time, but folks are very concerned about the length of time, and the uncertainty in licensing these digital application in RPS and ESFAS.

Now to that end, from an industry perspective, we have developed a working group. That's the next highest level you can have at NEI from an industry perspective, and headed by a Vice President of Engineering Technical Services, Amir Sharkarami at Exelon. And we look forward to working with the staff on moving forward all these various issues. We identified I think it was five priority issues, one of which was research with the staff at that March workshop, so we want to take that list and start knocking it off and move forward, because there are a lot of folks out there that wold like to move forward with digital applications, RPS and ESFAS. Most of them say that I'll move forward right now to the extent possible with the controlling sides, with the non-safety related sides and the controlling sides. And wait until things get a little more stabilized in the regulatory front until we know more of what we really have to do. What do we really have to do to have a quality submittal, and have a good timeliness in that application, but we're going to work on that with the staff.

CHAIR APOSTOLAKIS: Good. Thank you. Any other comments?

MR. KEMPER: Yes. And just to reinforce, just give of a good segue way, we're listening and taking serious exactly what the industry is telling us. I just received a user need to accelerate research in the area of diversity and defense-in-depth, and also advanced control room design issues, which is primarily prompted from that meeting that Tony just spoke to a couple of months ago.

CHAIR APOSTOLAKIS: Very good. Thank you, Alex. Let's close by going around the table and see what impression people got today. You want to start, Tom.

MEMBER KRESS: Sure. Well, I believe I saw a lot of progress since our last meeting. And I think the program is on the right track. Early on I was very skeptical that we could ever develop software reliability failure rates, but now I'm more hopeful. I think I see progress in this area. I'd like to second your comment, George, that it would be nice to have some early on judgments as to which systems actually need to be modeled, and what process one would use to model those particular ones. And I think risk-importance measures would be very useful there. No use to waste time on things that are not really risk-significant. And even though we don't have failure rates, I think you have to develop risk-importance measures for systems.

One area that kind of bothered me a little is when testing revealed no failures over a range of coverage, I think there should be a statistical technique to estimate the probability of having a given number of failures, and that has to depend on the amount of the degree of coverage, so I thought that needed a little more work.

I was a little skeptical of having the ability to incorporate time-dependent failure rates into PRAs. I think we need to figure out how to work around that, or avoid it. That sounds like a real problem to me. At some point in our subcommittee meetings, I'd like to have a more detailed discussion on how the lambdas are developed from the 1 minus Cs. I'm not sure how that's done.

I appreciated the industry's comment that failures per demand would be more interesting than failures per hours of operation. I think that's an area that needs to be thought about. I don't know, it seems to me that replacing analogs with digital almost automatically decreases risk. I don't know if we could make such a blanket determination or not, but that's just a thought.

I would like to support, add my support to the industry's comments that on several areas. One, re-evaluating what we mean by defense-in-depth in digital INC areas. And I really like Alex Marion's suggestions on the industry peer review, and cooperative research. I'm glad to hear that that looks like a possibility.

Eventually, I think we'll need to have reviews of digital INC installations in new plants, which may not be LWRs, and I don't think the acceptance criteria will be the same as are in Reg Guide 1.174, and I think somewhere along - I don't know if these guy's role to do that now, but somewhere along the line, we'll have to think about how to deal with them in the newer plants.

All in all, I see lots of progress. I'm hopeful that this –- to me, clearly there's a need to incorporate digital INC reliability into the PRAs, so I'm glad to see this work.

CHAIR APOSTOLAKIS: Thank you. Mario.

MEMBER BONACA: Yes. I voice most of the comments that Tom made. I mean, I see a lot of progress. And, in fact, more than I thought we would see by this stage. The area of determination of which digital system need to be modeled and what level of detail, that's an area, of course, of interest to all of us. But I think it's also important because it will define somewhat where you need to have dynamic modeling, and where you can stay with traditional methods.

I would be responsive to Mr. Gaertner's recommendation of not forcing incorporation of digital INC modeling in PRA. I mean, there may be other ways to do that. I would view the approach the NRC is choosing as one that they are choosing for their own independent validation and verification, but it is not the only way to go about that. And really, I believe there should be collaboration with industry very much at this stage. I think a collaborative effort can only be helpful.

I still believe there is a lot of technology out there available, at least some of it we saw ourselves when we went to Germany, and so there is a lot of experience that can be brought to bear, and from which we can really derive benefit, both from a regulatory standpoint, and from an industry standpoint.

Regarding Reg Guide 1.174, I mean, I'm -- I can see as work in progress so, of course, all of us have high expectations of that reg guide, because we are all supporting risk-informed regulation in this area, too. So that's pretty much my comments.

CHAIR APOSTOLAKIS: John.

MR. HICKEL: Well, this was my first foray into what your subcommittee had been doing, and I did appreciate the two letters I think you've shown me what they have done in the past. So I guess my perspective is really of maybe just a fresh set of eyes looking at what you've been doing already.

My immediate thoughts are that needs to be a more focused prioritization of where the staff is trying to develop modeling, and analysis capability. I don't know why the focus was on digital feedwater control systems. I would hope that there is some opportunity to get from the people in NRR that are maybe the users of the research efforts and the reg guides, like a picture, in the next six months we're going to have to review this, in the next two years we're going to have to review that, and five years out we've got advanced reactors, or evolutionary plants where we're going to have to take a position.

I would think that there is a need to have more ability to project and evaluate trip systems and ESFAS logic systems than was discussed here today. I think that's my first comment. My second comment is that I think that the data mining efforts that are going on right now on the Brookhaven research project, they appear to be more evolutionary. There's clearly a lot more data out there. I think there are better ways of getting it, but I think one of the things that I see that's out there is issues of configuration control afterwards, because these are the failures that clearly are occurring. Somebody gets a bad data set and they put it into all channels of the trip system, and that's not digital. You can do the same darned thing in an analog, an old analog system, but it's out there, and trying to understand those kind of controls, I don't think we're focusing on that.

There is a lot of experience that people have done that. There's a lot of experience out there from the LER system that there have been problems in calibration that result in people putting the wrong numbers into all channels, and they're assisted and guided by computer programs that are doing that for them.

Those kind of things are happening. This is not a highly complicated software reliability issue. This is just that people are following procedures, and on some occasions don't follow the procedures, and they put in wrong numbers into everything. And that issue is probably more likely to occur than some very highly unusual common cause hidden software failure. I'm thinking that the LER database can give you better estimates of that thing versus some unknown, undetected common cause failure of software.

I think the numbers can be extracted, and I do believe they will help better focus the efforts towards coming up with regulatory guidance that will be traceable back to history, and numbers, and be better focused. And I think those are the two main comments I'd have.

CHAIR APOSTOLAKIS: Okay. Thank you. I think I was pretty vocal all day. I still -- I just want to repeat that this issue of transition rates is something that I really have to understand better, what is the basis, and what do they really mean. And I think we're making a lot of progress, as I said earlier. Now we're discussing context, we're getting into it more deeply, what does it mean, and all that, and I'm confident we'll get some good answers soon. The issue of zero failures, I mean, we're fixing them all the time, and this paper, by the way, that was cited in the report from the IEEE transitions, was a pretty powerful mathematical analysis of what you do in those cases. I'm not saying we should do that, the mathematics is there.

So I'm very pleased myself with the progress that has been made, and I'm also happy that you guys are so willing to come and talk to us about things that are still evolving, but that's the whole idea of these meetings. We've tried it with 1.174 several years ago, it was pretty successful, so we're doing this now. And I also am very pleased that the industry decided to come and voice their concerns and ideas, because this is really what will lead us to something useful eventually. So with that, unless somebody has something to say, from the staff, the public? Thank you all very much. This meeting is adjourned.

(Whereupon, the proceedings went off the record at 5:16:33 p.m.)