Intel IDF – 3.7.06 Justin Rattner Keynote Presentations

[Video.]

Female Voice: Ladies and gentlemen, please welcome Justin Rattner.

[Applause.]

Justin Rattner: Welcome to the Intel Developer Forum. We're delighted to have everyone here this morning. It's just great to see everyone here bright and early and ready to get started for what is going to be a truly memorable IDF event. For you first timers, I'd like to extend a special welcome to IDF. You couldn't have chosen a better one to attend. I think this will be one of the most memorable IDFs in a long time. For you many alumni, those of you who've been with us for many years, I also want to extend a heartfelt welcome. The IDF team has put together one of the most outstanding programs we've seen in a long time, and I'm sure you'll enjoy it.

One thing we all have in common, whether we're developers, managers, industry analysts, or members of the press is we all love technology. I mean that's why we come together. Imagining and creating technology is what we're all about. And it's something we've done very well over the years. And IDF has really been a catalyst for bringing us together and pushing technology forward. That's why we started IDF in 1997 with this idea of bringing the community together. And that's why it's still going strong 10 years later. Yes, this is the 10th year for IDF. Now knowing what I know, as CTO, I can't think of another company I'd rather be a part of. Our products and our technologies that will be coming to market over the coming months are truly exciting. Even though we've been under tremendous

competitive pressure, and you might think we may have lost a certain enthusiasm for what we're doing, that's far from the truth. We are tremendously excited about what we're doing. And I believe when you learn about these technologies this week at IDF, you'll truly understand why I love my job.

And because this forum is for you, we've been listening very carefully to your feedback. Yes, we really do read all of those IDF evaluation forms you fill out after these sessions, in the classes and throughout the forum. And with that feedback, we've put together a program, which reflects exactly the requirements you've given us. There will be more opportunities to learn from the experts. You'll have more access to Intel's key technologists. We've even created an opportunity to go deep on some of the key topics of the day. We have Technology Insight sessions for the first time. There are four of them, which will be taught by the group CTOs and a number of Intel fellows. So you'll get some of our best and brightest folks sharing with you their deep technical knowledge in these key areas. It's my turn this morning, in the next half hour, to give you a preview of this broad landscape of technologies that are coming to market over the next few months.

Energy is on everyone's mind. It's the next frontier. We hear about it every day. You can't listen to the radio, you can't pick up a newspaper, and you can't browse the web without hearing about energy. Not only has it become a critical concern in our daily lives, it's become a critical concern in just about every platform we deal with. Whether you're talking about a high density rack-mount server or a sleek and quiet Viiv platform sitting in your living room, or a thin and light notebook

that you're carrying with you on your travels, even in a handheld device – each one of these has critical energy concerns. It may relate to battery life or relate to power density in the data center; but each one defines a power envelope, exists within a power budget that has to be understood and managed.

Now there's a fundamental tension here between performance and energy consumed. And that's what we're going to spend quite a bit of time talking about this morning. It's a classic tradeoff. It's one, for example, that automotive designers and builders have to deal with. We can either build a car that has tremendous acceleration and top speed and pay a significant price in terms of fuel consumption – the energy required – or we can build a very economical vehicle that has tremendous range, is very thrifty on gasoline, but whose performance is less than exciting. So that's a fundamental tradeoff and it's just as true in the information technology space as it is, say, in the automobile space.

So one might ask, is there a way out of this dilemma between power and performance? And we began to study this some years ago and the trend was a bit alarming. Over the years, beginning with Pentium back in 1993 and continuing to the present day with Pentium IV, every increase in performance required an attended increase in energy expended. Now this graph you see here, I want you to pay close attention to. I think it's the first time we've actually shared it publicly. What we're looking at here is relative performance on the horizontal axis versus energy per instruction on the vertical axis. We have factored out the effects of process technology on performance to

provide a comparison at the microarchitectural level. So we're actually quantifying how much energy is required to execute one instruction on each of these processors. As you can see, over this span of time, the amount of energy required to execute a single instruction has increased significantly – well over a factor of four over this time frame. So you might say, "Well, the data doesn't look very good." In fact, the trend seems to be in the wrong direction towards less energy efficiency.

But a few years ago, at the Israel Design Center of Intel, a team of engineers were assigned the task of creating a mobile processor that would offer excellent performance, but be very thrifty in terms of energy. Their initial effort, what you know as Pentium M, was remarkably successful at achieving this goal. In fact, it was so successful that it actually matched the energy efficiency – it had the same, if not better, energy efficiency than the original Pentium processor. So we went from this steep downward trend, in terms of energy efficiency, taking, in some sense, a giant step backward to the kinds of energy efficiency we got from processors more than a decade ago.

Succeeding generations of the Pentium M continued this trend, so even though they were getting faster, they weren't consuming more energy. The energy per instruction, as you can see from that line, has remained constant. Well you all know, those of you who attended engineering school, that two points define a line, and to us this meant, hey, these guys are on to something. They figured out how to build processors that can be quite competitive in performance but exhibit the same energy efficiency as the original Pentium. And that was really a

stunning result. Core Duo took this progress one step further and for the first time-achieved performance that was comparable to best in class.

So we had the foundation, we had the technical insights required to really revolutionize the industry. Let me show you some of the impact that this technology is having on design already. Of course Core Duo was created primarily to serve mobile platforms. Here's a Centrino Duo platform featuring the Core Duo processor. This one happens to be from Sony, a particularly nice one. But Core Duo, because of this unique combination of energy efficiency and performance is finding its way into other kinds of platforms. Here are two really stunning Viiv platforms, very compact, very energy efficient, cool, quiet, and long-running. The technology is so compelling that it's even found its way into blade servers. These are Xeon LV processors, the two of them sitting here, which are derived from the Core Duo architecture. And you'll hear more about this design in Pat Gelsinger's keynote that follows mine. So the Core Duo proved what is possible. And it's given us the foundation for going forward.

Now, I should say that the tradeoff between power and performance is not quite as simple as I just described it. There is really a more complex tension between three different elements.

Justin Rattner: Of course there's performance, and there's energy efficiency, but there's a third point on this triangle, and that's capability. This is the real challenge. How do we incorporate all of the capabilities of our leading-edge microprocessors into an energy-efficient design? Can we

add things, like 64-bit addressing, like virtualization, like management and security, without losing energy efficiency – and hopefully, while continuing to increase performance? That's the critical tension. Is it really too much to ask, to have your cake and eat it too? Well, we think the answer is, resoundingly: "No, it's not too much to ask."

Today we're announcing the Intel Core microarchitecture, as the culmination of this long period of research and development that brought us to this fundamental insight into microprocessor design. Core microarchitecture combines energy-efficiency – the same kind of energy-efficiency that I've just shown you in the Pentium M and Core Duo lines – with the features and capabilities that are now expected in the top-of-the-line microprocessors. Full 64-bit capability, virtualization, and all of the other sophisticated features of these high-end machines. Together, they deliver outstanding performance. Not just in mobile platforms (where the technology originated) but, in fact, across the entire range of platforms. So you'll see Intel Core microarchitecture again in mobile systems, but significantly now moving into desktop and server systems.

Of course, every great microarchitecture starts with great semiconductor technology. That's one of the things that we're most proud of at Intel. All of our new core microarchitecture-based processors will be fabricated this year in our high-volume, 65-nanometer technology. This is a technology that's proven in volume production at two Fabs around the world today. It's a terrific technology for designing a next generation microprocessor. It gives us 20 percent better transistor performance, and it gives us 30 percent

lower switching power, which is key to maintaining that energy efficiency metric that I mentioned earlier.

We believe we're over a year ahead of the competition in delivering 65-nanometer technology, and we'll continue to invest in that technology, and bring two more Fabs online this year. Starting in the second half of next year, we'll move to 45-nanometer technology, and we're making excellent progress here. You may have seen the recent announcement of the 153-megabit static RAM fabricated in our 45-nanometer technology. This is usually the first and major test vehicle in a new technology, and this device has given us great confidence in the success and the promise of 45-nanometer technology. It'll give us bigger caches; it'll give us smaller cores; it'll allow us to put more cores on a single die. I'd like to call your attention to one of the Tech Insight sessions that'll be taught by Intel fellow Paolo Gargini. He'll go into great depth in terms of process technology and the evolution of Moore's Law.

Now, there are hundreds of innovations in the core microarchitecture that are incorporated in this family of processors that I've just described. But I want to take this morning, in the limited time that I have, to really highlight five key ones. They're all involved with balancing performance with greater energy efficiency. We continue to have to make that trade-off.

The first of these are changes to the instruction pipeline itself. We've widened the execution pipeline, so we can complete four instructions in a single clock. That's given us the ability to get more done in a

single cycle, and by doing it in fewer cycles, we actually consume less energy. We've continued to refine the 14-stage pipeline, evolving from Core Duo and adding to Micro-fusion, the ability to combine pairs of microinstructions into a single microinstructions, and execute that in a single cycle in the pipeline. We're announcing today the addition of Macro-fusion, where we can actually take two Intel architecture instructions – now, I'm talking about high-level instructions – and combine them into a single instruction, and execute that in a single clock. For example, compare and jump becomes a single instruction in the pipeline, and is executed quickly and efficiently.

The next feature I want to talk about is the improvements to the SSE family of instructions. Some of these instructions, in the past, have executed in a single cycle, but not all of them. Today we're announcing that all of the processors featuring Core microarchitecture will execute the entire family of SSE instructions in a single cycle. This is going to offer important performance improvements for those applications in video, in audio, in digital photography, and wherever media processing is required.

Now, to minimize memory traffic, we've incorporated an advanced L2 cache architecture. Not only does this cache structure feature reduced power levels, it provides significant improvements in the access time to cache, so the processors spend less time waiting, and it allows for the efficient sharing of information between the processors. So when they have to exchange data – let's say you have two threads, each one running on one of the cores, and they have shared data, and they need to exchange that shared data between them. It takes place efficiently

and quickly in the shared cache structure. We've even gone to the extent of giving the cache the ability to serve one processor in its entirety. By that I mean that, if one processor should happen to go idle (which certainly may occur in a mobile platform, where we may just be waiting for the user to click the mouse), the other processor has access to the entire cache. We're not partitioning the cache in some way. It's not private to an individual processor. That full L2 cache becomes shared across all the processors.

The next capability is an improved set of pre-fetch algorithms. Previous generation processors have used pre-fetch to bring data into the cache in anticipation of what the cores will need. We've developed an improved set of algorithms that will make that significantly more efficient. And beyond that, we've added additional flexibility in the way loads and stores are ordered as they're sent to memory. We're now able to bring loads above stores. Now that can be dangerous if the loads and the stores are touching the same memory location, but through hardware techniques, we're able to detect that conflict and resolve it, and allow those instructions to proceed. So that gives us a better capability of ordering loads and stores, improving energy efficiency, and improving performance at the same time.

Now designs such as this have to manage power aggressively, and we've used the technique known as power gating throughout the Core microarchitecture. Power gating lets us shut down various portions of the chip, sections of logic that aren't needed at a particular instance to support instruction execution. And we've done this quite extensively. And another one of the Technology Insights that I mentioned earlier,

this one hosted by Senior Fellow and Digital Enterprise Group CTO Steve Pawlowski as well as Intel Fellow Ofri Weschler, they'll go into this kind of capability in much greater detail. And you'll have the opportunity to learn exactly how we implement these sophisticated forms of power management.

Well, okay, nice set of technical features, but what does it really mean? What's the ultimate impact of all this technical sophistication on developers and users? Let me start with the Merom processor for mobile systems. Here we expect to see a very handsome gain in performance over Core Duo systems while retaining the same great battery life. So again, we're staying on that energy efficient line I showed you earlier, consuming just about the same amount of energy per instruction, but delivering more performance in mobile platforms and giving you excellent battery life. It gets more exciting as we move Core microarchitecture to the desktop. Conroe is the desktop processor based on core microarchitecture, and it features more than a 40 percent improvement in performance with a 40 percent reduction in power. Now you're really beginning to get a feel for the capability, the potential of this new microarchitecture -- energy efficiency and high performance.

Well like the man says, you ain't heard nothing yet. Woodcrest is the server processor based on Core microarchitecture. It sees an 80 percent improvement in performance and a 35 percent reduction in power. So this just seems to get better and better as we look at each class of system. You'll have the opportunity to see these systems in operation. As I said, Pat will be here in a little bit to show you these systems.

You'll see the actual performance. You'll see the actual power levels. You'll get a first-hand glimpse at the potential of this new microarchitecture.

Now there's one important characteristic of all the processors implemented in Core microarchitecture, and that is they're all multi-core machines – all multi-core processors. And you might ask yourself, well, if Intel is able to make these tremendous gains in energy efficiency and performance, why is it bothering with multi-core? Seems like its making things unnecessarily complicated. And since I get asked that question so regularly in my travels, I figured it was a good opportunity this morning to share the reason with everyone here.

So let me start – this is a simple example – and explain why multi-core is so important and such a key part of energy efficient performance. Here's a typical processor, or we'll declare it the unity processor – everything is one about this. So one unit of power and one unit of performance. And we'll assume that it's operating at its maximum frequency, given the underlying semiconductor technology. Now this will set our power budget for the rest of this explanation. Now you might say, well, we could make it go faster by raising the frequency. Let's say we did that. Let's say magically, regardless of the technology limitations, we were able to up the clock frequency by 20 percent. You know, what the tweakers do. We're going to over-clock the design. So here's what happens. Frequency goes up 20 percent. Performance doesn't go up as much as that, because there are other issues to consider. We have memory access times that aren't scaling with

processor frequency. But the significant change is the jump in power consumption. And here you see, this is the right-hand bar, power has gone up 73 percent. So we've seen, for a very modest increase in performance, order of 13 percent, power is up 73 percent. That's not a very good tradeoff and that's one of the reasons why we just don't turn the clock up because it costs us tremendously in terms of energy efficiency.

Now the alternative to this, which may seem surprising at first, would actually be to turn down the clock. Let's say we were to turn down the clock 20 percent, now what happens? As you can see here on the graph, the power drops by nearly 50 percent, but performance is only down 13 percent. So for this dramatic reduction in power by simply lowering the clock frequency, we get a relatively modest decrease in performance. But more importantly, it opens up the power envelope. Now we have room under that unity processor power budget to add a second core. This is exactly what happens. With a second core, power comes up just about to the same point – a little bit above – where we were with our baseline processor; but performance is now up 73 percent. This is the fundamental reason why multicore is a key part of energy efficient performance. You can't get these kinds of performance gains at this kind of energy level any other way.

We'll be continuing down this path in the coming years. You may have seen our announcement of the first quad core processors. You'll hear more about them as well at this IDF. So the question is: are we going to double the number of processors again in 2008? Well, probably not. I think Intel is taking a relatively conservative approach here focusing

on single processor performance, single thread performance, because that's where most of the applications are today, and adding processors to take advantage of threaded applications as they come to market and as they get developed. So you won't see mediocre core performance simply for the sake of getting more cores on the die. As the community responds to multi-core architecture and those applications develop, we'll roll out additional cores.

Of course, the key development required for exploiting multi-core processors is multithreaded software. This hasn't gone unnoticed at Intel. In fact, we have a major, major effort underway to support the development of multithreaded software – a whole family of tools and technology to assist you, the developer, in creating these applications. Additions to our VTune technology, our performance analysis tool, so that you can look at the performance of multiple threads and make sure you've got them properly load balanced. Or it may include thread-capable, thread-optimized, Math Kernel Libraries and Intel Performance Primitives. So we've sort of taken that work and done it for you by multi-threading the key libraries in advance of your need.

Now many leading ISVs have already taken advantage of these technologies and these programs are well under way to delivering multi-threaded software. I really hope that at this IDF, no one goes out of here – certainly no software developer goes out of here – not committed to bringing multi-threaded technology to their products and their systems. The tools are there. The technology is there. The training is available here at IDF and elsewhere. It's really time to get on board the multi-threaded train.

So you might think, okay great new core microarchitecture. Wow – some amazing performance figures and incredible energy efficiency. Multicore processing makes sense – a great way to add to performance while retaining energy efficiency. And finally, multithreaded software is happening. The tools are there, people are learning how to do it, and the results are very encouraging.

Have we left something out? Have we forgotten something? You may have noticed one word has not passed my lips – platform. That's the next opportunity for all of us. Taking a look at the rest of the platform and seeing how to extract yet more energy efficiency from the platform level. This diagram shows you today's power split – the typical server – between the platform and the processor, it's nearly 50/50. What happens when you bring Core microarchitecture to bear on that typical server? Well, here you see the problem, the challenge, the opportunity. Processor power is now only a third of total platform power, while the rest of the platform is dominating the energy consumption. This is an opportunity for great improvement. Let's look at a typical system that's operating in idle mode. This would be common in mobile systems, obviously, but it could be quite useful in servers and desktops. So here the processor has entered a low-power state, an idle power mode, but the rest of the platform has to stay up.

Now why does the rest of the platform have to stay up? Well two reasons. One, the display has to be refreshed. Even if the image on the screen is not changing, we still have to keep refreshing the display. That's the basic characteristic of this architecture. Also the operating

system is periodically waking up. We say it's responding to ticks. It wakes up every few milliseconds to service any devices that may have required attention in the interval that it was shut down. And that brings the platform up to full power.

What if we could change the platform in such a way as to keep most of it, or more of it, at idle for longer periods of time? One example of doing that is to create something called Display Self Refresh, so that the chipset and the processor power-down and add a small amount of memory, only that required to refresh the display is left running. We can also modify the operating system to use a variable tick, to only wake up when it's absolutely necessary. It may require some changes to the way devices behave and certain interfaces are defined, but now the operating system is bringing the system up only when it's absolutely necessary.

We've actually created an experimental system you see here. The experimental system implements exactly the capabilities you see on the screen. This is our Silverton platform. Believe it or not, this big motherboard is actually the electrical prototype for a hand-top system, an ultra-mobile PC. Clearly this is not the mechanical prototype, or it would be vastly smaller. And here's the display subsystem with the self-refresh capability.

So I'll back up here so we can see the power meter running. The platform's operating at about 6.5 watts, which is still more than we'd want to see in an ultra-mobile device. Now I’m going to ask Doyle to enable this extended idle mode that I was just describing. It will take

just a moment. And then you'll see the power begin to drop substantially after a few transients. So now we're in extended idle. The screen still has the image. That looks like Robby and me on stage last fall. And we're running at 3.5 watts. So we've dropped from 6.5 watts to 3 watts. A 3-watt power saving in the platform, just about 40 percent in the platform as a result of implementing extended idle mode.

Now I want to make one point very clear, that we haven't lost any responsiveness in the system. So if the user was to move the mouse, you'd see it immediately come out of extended idle mode and begin processing those mouse clicks. So from a user point of view, no change in system behavior, even though battery life, for example, in a mobile platform would be greatly extended.

Now to take this one step further – I'm going to turn off the motherboard.

Justin Rattner: All right, so the power is now off on the motherboard. We've kept power running to the display subsystem, if you want to disconnect the cable to prove that there is no slight of hand here. And the picture of me and Robby continues on the screen, and most significantly, the power is now down to only 1 watt. So the potential for managing power at the platform level is really great, and it's something we need to address and take the opportunity to improve in all of our systems.

So here we are, looking at a new generation of energy-efficient systems, harnessing this technology, delivering this technology to our

customers is going to require tremendous cooperation, particularly at the platform level. It's not just an operating system issue, although there is operating system work to do. Not just an issue of IO devices, communication storage and display. It may even include some standards issues. We look at the standard interfaces, and we see that some of them actually require the platform to wake up to service various device needs. I think this is something we can all come together as a community and really address, and really deliver benefits to the end user.

So here we are, at the dawn of a new age of energy-efficient performance. It's really our job to imagine, and our job to create and deliver this technology to the industry. The opportunities are great. The challenges are great also. But we, together and collectively, are the agents of change. We can help make energy-efficient performance a reality for the industry. It's our job to do it, and I want to encourage everyone this morning to immerse themselves in the technology here at IDF, to learn what you need to learn, and really have a truly memorable experience at IDF. Let's all leap ahead and have a great Intel Developer Forum. Thanks very much.

[Applause.]