The Next 25 Years in EDA: A Cloudy Future?

Leon Stok
IBM Systems and Technology Group

Editor's notes:
This paper is based on an invited talk presented at the 50th DAC. It discusses the trends and challenges in design and EDA industry with special emphasis on the unique opportunities in cloud computing.
—Yervant Zorian, Synopsys

At the 50th Design Automation Conference, Soha Hassoun organized a session to review the first 50 years of EDA and predict the next 25. While Bill Joyner [1] could draw upon the vast paper resources that MPA collected during the first 25 years of DAC and Rob Rutenbar [2] was able to apply some amazing analytics to the second 25 years of DAC proceedings, I turned to crowd sourcing to solicit input on the next 25 years. It was clear from Bill's and Rob's presentations that many people were drawn to the EDA industry for the same reason I got hooked: EDA was on the forefront of computing. Many of the largest computational problems that needed to be solved were EDA problems. The university groups focusing on EDA had the largest and fastest computers. If one wanted to solve large-scale computer science problems, EDA was the place to be for most of the last 50 years. Is that still the case?

The replies to the "next 25 years of EDA" survey pointed out many of the challenges that EDA tools will need to address in the next decade. Key challenges will come from next lithography solutions such as Triple Patterning, EUV, DSA (direct self-assembly) or Ebeam. Many respondents pointed to the problems that next generation nanodevices such as photonics, bio based circuits, quantum, and carbon might bring.

However, as shown in Figure 1, a large number of participants pointed out how IT and specifically cloud computing will have a major impact on the way EDA tools will be used. This is particularly interesting because in the "EDAC Forecast Meeting" on 14 March 2013 [4], the impression was left that cloud and EDA do not mix well. Earlier DAC panels [5] also pointed out the significant hurdles to overcome for EDA to be cloud friendly. In this article, I decided to zoom in on this one aspect of the next 25 years of EDA and will outline why we need to bring Cloud Computing and EDA together to deliver the next leap of design productivity and at the same time, restore the excitement in the EDA discipline.

Key data points

There are several key trends that will affect the electronic design environments of the future.

NSF workshop observations

The report on the 2009 NSF Workshop that focuses on EDA's Past, Present and Future was published by Brayton and Cong in two parts in *IEEE D&T* [6]. The second part outlines the key EDA challenges. Interestingly, it has a only a few challenges printed in bold: **intuitive design environments, simplified user interfaces, standardized interfaces, and scalable design methodologies** all leading to **disciplined and predictable design**. Very few of these challenges require drastic innovation in the key algorithms underlying the EDA tools. Instead they all point to the design flows and the design

Digital Object Identifier 10.1109/MDAT.2014.2313451
Date of publication: 04 April 2014; date of current version: 19 May 2014.

2168-2356/14 © 2014 IEEE

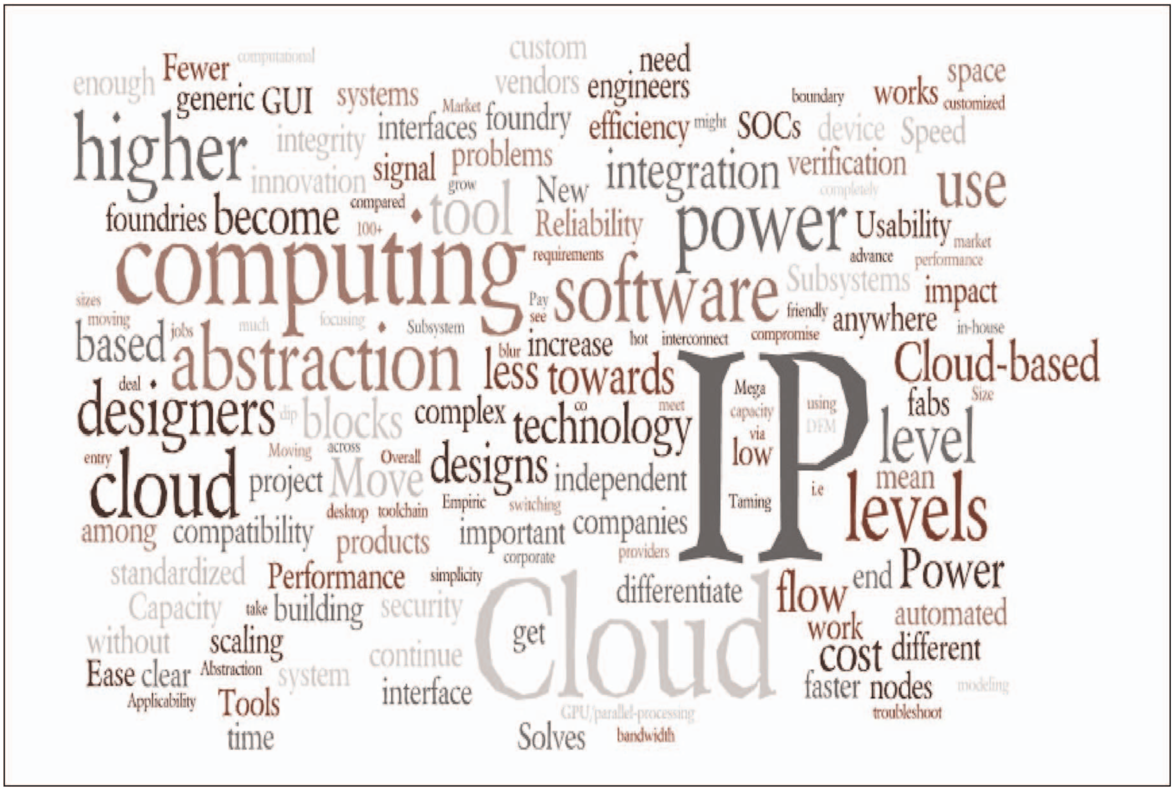

Figure 1. What will affect your Design Environment in the next 25 years? The size of the words is determined by the frequency of the word in the response to the survey by the 120 respondents.

environment through which the designers interact with design tools and to the scale of problems that need to be solved. These challenges largely overlap with the areas Big Data cloud applications have been focusing on and made tremendous progress in. One can certainly argue that a Big Data application like Google maps has a simple and intuitive user interface, has standard APIs to annotate data and has been architected to be very scalable. Before we look at what these technical advances could mean to EDA, let us look at some data to see what a current large scale EDA design environment looks like.

Large-scale design environments

Most EDA tools are deployed in complex design flows and used by a large number of designers. Many tools communicate with each other through arcane file formats [7] and produce data in huge text based log and report files. Most tools have a very large number of configuration parameters and require a very elaborate setup for a particular technology or design style, usually reflected in elaborate and complex long control scripts.

It is not uncommon for design teams to write 1–2 million of lines of Skill or Python scripts to create library cells and IP blocks. A complete design and verification flow can quickly add up to 1–2 million lines of TCL to control the tools and deal with the setup and environment in which IP and models are stored. And once the design flow is up and running, a large number of Perl scripts are written to extract the key information from the terabytes of reports and design data. It is often difficult to know how many of these scripts exist in a design environment since they are often owned by individual designers. In addition to all the configuration and control files a large amount of data gets generated and stored.

Let us look at the amount of data produced by a design team designing a 10 B+ transistor processor in 22 nm technology as shown in Figure 2. It takes about 12 Tb to store the entire golden data (including incremental revisions). Logic and verification takes

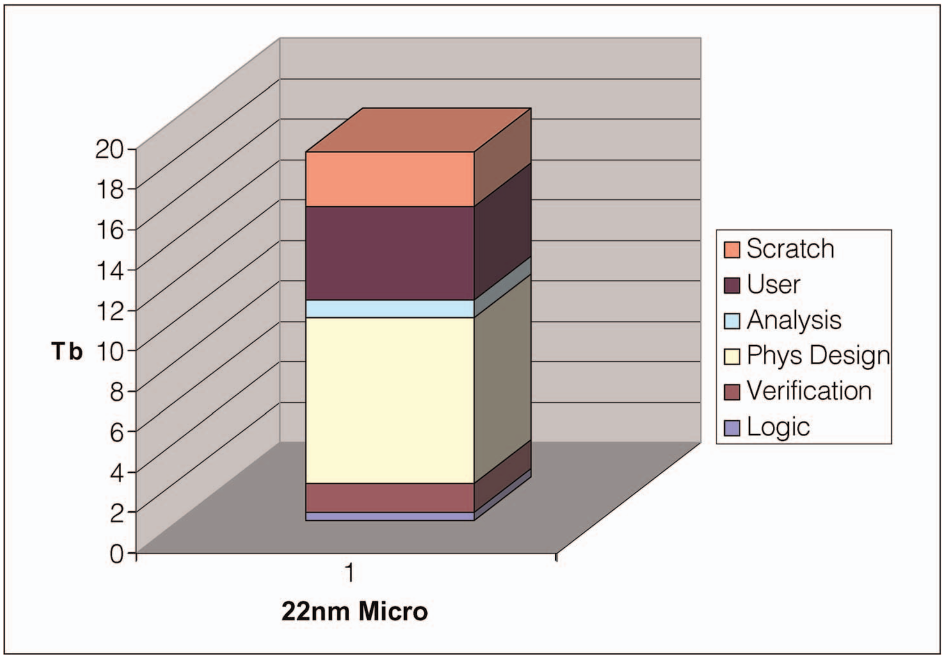

Figure 2. Design data for a 10 B+ transistor 22 nm chip.

about 2 Tb, the physical design data about 8–9 Tb and another 1–2 Tb for analysis reports. In addition individual users keep another 6 Tb of local copies in user and scratch spaces.

After the design is finished, it gets compressed and streamed out to about 3 Gb of OASIS. The product engineering team blows this up to around 1 Tb during mask preparation operations. Finally, another 5 Tb of test and diagnostics data gets generated in the postsilicon process.

This looks like a significant amount of data but it tops out at 20 Tb for a multiyear design and manufacturing project. While this was once a phenomenal amount of data that put EDA at the forefront of storage and compute needs, today, it has been surpassed by many large cloud-based Big Data applications. Let us look at some of them.

Data sizes

Let us try to compare the scale of some of the cloud applications with the problem sets we deal with in EDA. We will take a look at Google maps [8] and some key metrics that we can compare with the design data above.

1) **How much data has Google Maps accumulated?** Combining satellite, aerial and street level imagery, Google Maps has over 20 petabytes of data, which is equal to approximately 21 million gigabytes, or around 20,500 terabytes.
2) **How often are the images updated?** Depending on data availability, aerial and satellite images are updated every two weeks. Street View images are updated as quickly as possible, though Google wasn't able to offer specific schedules, due to its dependence on factors such as weather, driving conditions, etc.
3) **In the history of Google Maps, how many Street View images have been taken?** The Street View team has taken tens of millions of images since the Street View project began in 2007, and they've driven more than 5 million unique miles of road.

How does this compare to the design data? Comparing the 20 Tb/design versus the 20 Pb of the fully annotated Google maps, the design data is only 1/1000th the size. A large chip has 5 km of wire compared to the five million miles of road to accumulate street view images. The scale of intersections on the chips are in nanometers and the road crossings are in meters, but the street view data annotated is significantly larger that the physical design data needed to be associated with each stretch of wire and via intersection.

One major difference is that the core EDA design data is certainly more dynamic than the more static basic map of roads in an application like Google maps. EDA tools can much more quickly reroute wires than physical roads can be built. Let us look at another data point of the velocity of data in a cloud application like Twitter. About 12 Tb of Tweets are generated each day. While we have no accurate data on how much of the design data changes each day, since it tops out at 12 Tb after a multiyear project, it is safe to assume that only a small fraction of it changes daily. While the EDA data is more highly connected than the many independent tweets, many modern analytics applications search for and create large number of connections in seemingly independent data.

Big Data spans four dimensions: Volume, Velocity, Variety and Veracity. As shown above, EDA's data Volumes are small compared to Big Data applications. 12 Tb of tweets a day, or the analysis for specific patterns in 500 million daily phone calls exceed the Velocity at which most EDA flows produce data. EDA data is of much less Variety than the much more unstructured data collected from many sensors in Smarter Planet applications. And as far as Veracity goes, 1 in 3 business leaders do not trust the information they use to make decisions, while EDA has developed very reliably sign-off tools to drive major tape-out decisions. Most of the modern day Big Data cloud infrastructures can easily scale to EDA sized problems and beyond. From the above, we conclude that most of the modern day Big Data cloud infrastructures can easily scale to EDA sized problems and beyond.

Why has EDA not taken off in the cloud?

The discussions around cloud and EDA [5] have focused on security and licensing models. Indeed, only when sufficient security guarantees are given will designers put their entire IP on a public cloud. However, EDA flows can run in private or hybrid clouds, taking full advantage of the massively distributed Big Data software infrastructure without the security issues. In the meantime lots of progress is being made on security in many cloud applications. Security issues will likely sort itself out in the near future, driven by lots of other sensitive data that is moving into various clouds.

The licensing cost is a business model discussion. There are very few software industries where users pay $50 K per user per year for access to some software tools. In addition, most EDA users actually pay for the licenses they are not using. Most large corporations have pools of licenses tuned towards the upper end of their usage. The licenses are available to ensure that design teams do not get stuck in critical peak usage times. Any switch to a more pay-per-use model will require significant changes to the EDA industry business practices. Instead of paying for tools one does not use, one will be paying for only those they use. It will require someone to figure out how much to charge for each run. What if the tool does not produce the desired results? Does the user still need to pay if the tool does not finish? What about the quality of the results? Does the user need to pay if the design tool does not produce a well-placed design?

Both the security and licensing will be sorted out if there are compelling reasons to run EDA on a large cloud. Can EDA tools take advantage of the cloud software infrastructure? Can we make the tools a lot more productive and designer-friendly and therefore unleash new business model opportunities? We need to take the next step in the tool evolution to get there. Despite the fact that EDA vendors call many of their tool collections "platforms", most are still stuck in the integration phase. Many tools originate from different acquisitions or different internal teams, with different internal databases, different coding conventions and different control script conventions. This has lead to a major integration challenge, which in Geoffrey Moore's [9] terminology is typical of the Application Innovation stage. Re-implementing EDA flows in a Big Data, cloud friendly way will unleash the creativity and growth associated with the next step in evolution: the Platform Innovation stage (Figure 3). What will it take to make that happen?

EDA in the cloud

What does a designer (the EDA tool client) really want? She wants to get to the DATA from anywhere and any place. She wants the DATA to be there without her waiting for it. She wants to analyze the DATA with whatever tools she can lay her hands on to

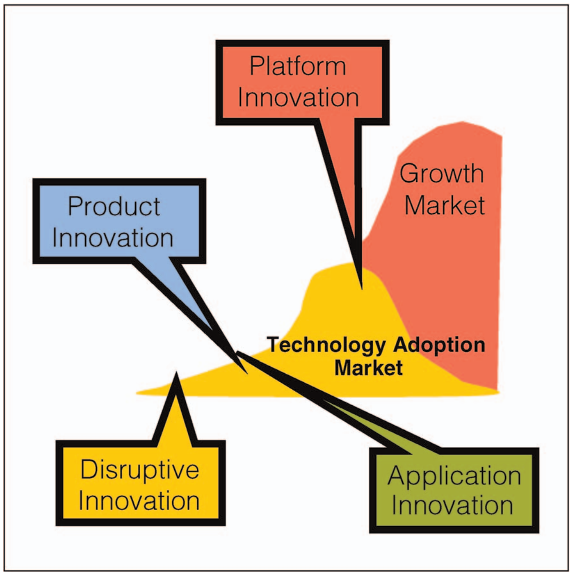

Figure 3. The Platform Innovation Stage.

learn how to improve her design. She wants to know how to get from A to B through the design process and wants design data, design navigation and a design flow to act like Google maps. For example, wouldn't it be great if understanding timing and congestion problems in your design is no more difficult than turning on traffic information in Google maps? Wouldn't it be great if we could annotate key manufacturing data from inline inspection tools just as easy as Street Views to our design data?

This type of rapid analysis and optimization can only be accomplished if the entire design data is in a (set of) live database(s) distributed among many machines in the cloud. When design changes are made, the live data needs to be instantly and incrementally communicated and updated. We know for example how to do this for timing analysis integrated with a place and route flow. However, this needs to be extended to all analysis. Analysis engines, running on several parts of the design, instantly recompute their information, and by the time the designer navigates to that portion of the design, the results are ready for him to look at. They produce the right abstractions that are needed by the higher levels of hierarchy in the design. Instead of thinking about synthesis, place, route and timing algorithms, this DATA-centric EDA paradigm will start from the data, map it to the cloud compute infrastructure and put the apps (e.g., placement, routing, timing analysis) on top of that using well defined EDA-OS APIs. Clearly there are some technical challenges here. EDA data is more connected than many of the social media applications. However, in many applications we have seen that EDA data viewed the right way is inherently more parallel than initially thought. The fact that tens or hundreds designers can work simultaneously and productively on a design makes it clear that the parallelism exists in the design process, albeit sometimes not in a single optimization run.

While EDA data is certainly more volatile during the optimization part of the process, the increased reuse of IP and increased use of hierarchy with appropriate abstractions has resulted in a much larger portion of the design data to be static in modern hierarchical designs. Furthermore only a small portion of the system and logic design gets (re-)done each day. With version control fully integrated in the data itself, the knowledge of what actually changed can lead to a whole new set of algorithms.

The key challenge is to find out how to lay out the design data effectively on top of the IT infrastructure and find the right set of APIs. How does the data need to be distributed among the disks, solid state memory and main memory? And how does this all need to be connected such that the physical network does not get in the way. The emergence of Software Defined Networking and Software Defined Data Centers will be a key asset and allow the IT infrastructure to be much more tuned to the task.

Much like Google and Facebook provide their own middleware on top of open source Linux and well-defined (proprietary) compute infrastructures, the EDA industry needs to ask itself the question: what can we do to make an EDA-OS run on top of an OpenStack and/or OpenCompute-like infrastructure to allow EDA tools to be written like apps that interface with live design data through well-defined APIs? Through SDN, standards are rapidly emerging that can make this effective and feasible.

The value is moving from the algorithms to the DATA. This will create a phenomenal opportunity for new business models in EDA and potentially solve the business model deadlock surrounding EDA and cloud. Gary Smith noted in his 50th DAC presentation: "Give away the tools, charge for the models." He might very well be right. There are many examples in the industry to look at. According to ProgrammableWeb [10], there are 10,760 published APIs on the web as of 12 January 2014. Gartner predicts that by 2014 75% of the Fortune 1000 companies will offer public APIs.

This will significantly lower integration costs, drive customer engagement and extend business models, offering new revenue opportunities for existing company assets. Companies pay each other for access to the data and access to the results. They do not pay a license fee to the analytics software that produces the results. Will there be a time when we pay for a timing report instead of licensing a timing tool? Will there be a time where we pay more for a better placed and routed result instead of licensing the physical design tool? John Musser [11] lists at least 20 business models that have emerged using APIs with many more rapidly following.

Gartner predicts that by 2018 50% of the cost of implementing new large systems will be for integration. Due to trends outlined above, the EDA industry has already crossed this point a while back. Lowering integration costs will be crucial to keep

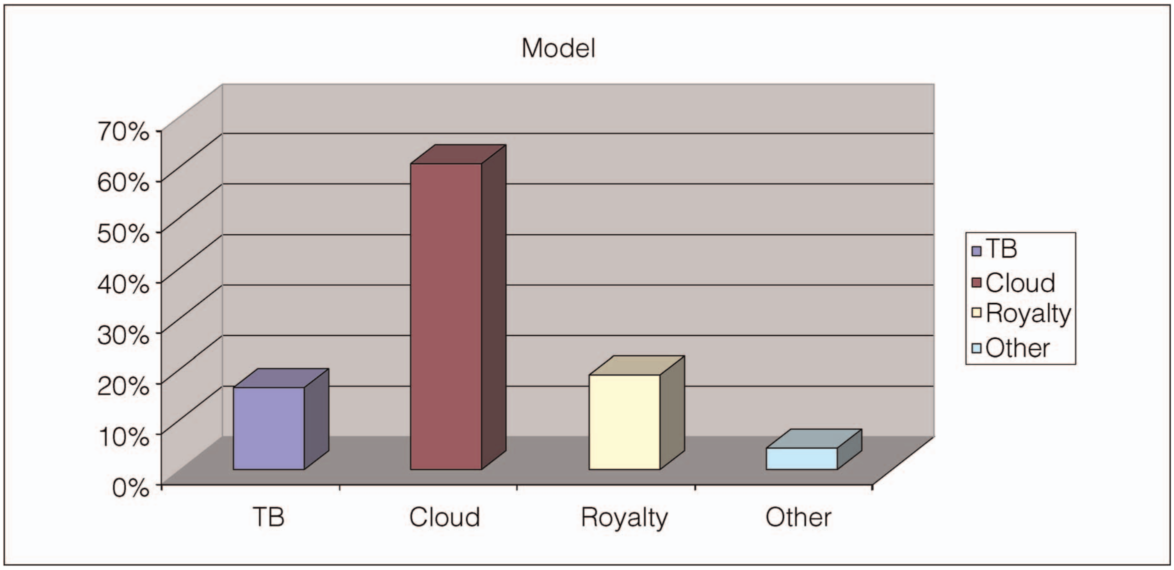

Figure 4. What will EDA tool licensing model look like? [TB = TimeBased, Cloud, Royalty (on IC production) or Other (mostly combination)].

EDA tools somewhat affordable while allowing investment in new opportunities. In addition, deployment of EDA tools in a controlled cloud environment can significantly reduce the support costs. EDA companies spend upwards from 25% of their resources on Support and Application Engineers. The likelihood that the EDA business model will change is also supported by the DAC survey (see Figure 4): 60% of the participants think that EDA licensing will move to some cloud model.

It is time for the design and EDA industry to embrace cloud computing. We need to look past cloud as an IT cost saving, but as a way to bring a significant higher level of productivity to the design community. EDA tools need to be constructed such that THE DATA is central in the EDA flow, not the optimization and analysis algorithms. Taking advantage of cloud technologies will drive to more intuitive and simplified user interfaces. Using cloud APIs will enforce standardized interface development. The massive compute power of a cloud-based installation will provide more scalable design solutions and better optimization. Analytics solutions will allow for a much better handling of the design reports than the current Perl scripts. Cloud-based EDA tool installations will be significantly easier to manage and result in much more robust and predictable design flows. We need to draw upon the many engineers that left electronic design automation and joined the numerous social media, analytics and cloud companies. Let us bring them back with their newly found expertise and apply it to electronic design. This will be necessary to take on the challenges to design large systems-on-chips built from future nanoscale devices.

Does EDA face a cloudy future? You bet! But cloud computing has given the word **cloud** a very positive connotation. Getting the cloud into EDA will bring EDA back to the frontiers of computing and create a wave of excitement that will attract new top talent. ■

■ References

[1] W. Joyner, "EDA: The First 25 years," presented at the 50th Design Automation Conference, Jun. 2013. [Online]. Available: www.dac.com

[2] R. Rutenbar, "EDA: The Second 25 years," presented at the 50th Design Automation Conference, Jun. 2013. [Online]. Available: www.dac.com

[3] Stok, "EDA: The Next 25 years," presented at the 50th Design Automation Conference, Jun. 2013. [Online]. Available: www.dac.com

[4] EDAC Forecast Meeting, San Jose, Mar. 14, 2013. [Online]. Available: http://www.edac.org/events/2013-EDA-Consortium-Annual-CEO-Forecast-and-Industry-Vision/video

[5] "Does IC design have a future in the clouds?" in *47th Design Automation Conference*, Jun. 2010. [Online]. Available: www.dac.com

[6] R. Brayton and J. Cong, "NSF workshop on EDA: Past, Present and Future," *IEEE Design and Test of Computers*, pp. 68–74, May/Jun. 2010.

[7] [Online]. Available: En.wikipedia.org/wiki/CategoryEDA_file_formats

[8] [Online]. Available: http://mashable.com/2012/08/22/google-maps-facts/

[9] G. A. Moore, *Dealing With Darwin*, Dec. 2005.

[10] Jan. 12, 2014. [Online]. Available: http://www.programmableweb.com/

[11] [Online]. Available: http://www.slideshare.net/jmusser/j-musser-apibizmodels2013

Leon Stok studied electrical engineering at Eindhoven University of Technology, The Netherlands, where he graduated with honors in 1986. He obtained his PhD degree from Eindhoven University in 1991. He worked at IBM's Thomas J. Watson Research Center as part of the team that developed BooleDozer, the IBM logic synthesis tool. Subsequently, he managed IBM's logic synthesis group and initiated the development of the first physical synthesis system: PDS, IBM's Placement Driven Synthesis tool. From 1999 to 2004, he led all of IBM's design automation research as the Senior Manager Design Automation at IBM Research. He is currently Vice President, Electronic Design Automation at IBM. He entered the field of Design Automation 25 years ago intrigued by the type of problems being posed by Moore's law. He has enjoyed working on problems from high-level synthesis to prescriptive layout design and DFM. In these 25 years, he attended most of the Design Automation Conferences, as a presenter of his original work in papers, a reviewer of the state of the art in tutorials or as a panelist to give his opinion on current issues. He served in many roles as a member of the DAC executive committee and as the chair of the 48th DAC. He is a Fellow of the IEEE.

■ Direct questions and comments about this article to Leon Stok, IBM Systems and Technology Group.