Toward a DNA-Based Archival Storage System

Storing data in DNA molecules offers extreme density and durability advantages that can mitigate exponential growth in data storage needs. This article presents a DNA-based archival storage system, performs wet lab experiments to show its feasibility, and identifies technology trends that point to increasing practicality.

James Bornholt
Randolph Lopez
University of Washington

Douglas M. Carmean
Microsoft

Luis Ceze
Georg Seelig
University of Washington

Karin Strauss
Microsoft Research

The "digital universe" (all digital data worldwide) is forecast to grow to more than 16 zettabytes in 2017.[1] Alarmingly, this exponential growth rate easily exceeds our ability to store it, even when accounting for forecast improvements in storage technologies such as tape (185 terabytes[2]) and optical media (1 petabyte[3]). Although not all data requires long-term storage, a significant fraction does: Facebook recently built a datacenter dedicated to 1 exabyte of cold storage.[4]

Synthetic (manufactured) DNA sequences have long been considered a potential medium for digital data storage because of their density and durability.[5–7] DNA molecules offer a theoretical density of 1 exabyte per cubic millimeter (eight orders of magnitude denser than tape) and half-life durability of more than 500 years.[8] DNA-based storage also has the benefit of eternal relevance: as long as there is DNA-based life, there will be strong reasons to read and manipulate DNA.

Our paper for the 2016 Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS) proposed an architecture for a DNA-based archival storage system.[9] Both reading and writing a synthetic DNA storage medium involve established biotechnology practices. The write process encodes digital data into DNA nucleotide sequences (a nucleotide is the basic building block of DNA), synthesizes (manufactures) the corresponding DNA molecules, and stores them away. Reading the data involves sequencing (reading) the DNA molecules and decoding the information back to the original digital data (see Figure 1).

Progress in DNA storage has been rapid: in our ASPLOS paper, we successfully stored and recovered 42 Kbytes of data; since publication, our team has scaled our process to store and recover more than 200 Mbytes of data.[10,11] Constant improvement in the scale of DNA storage—at least two times per year—is fueled by exponential reduction in synthesis and sequencing cost and latency; growth in sequencing productivity eclipses even Moore's law.[12] Further growth in the biotechnology industry portends orders of magnitude cost reductions and efficiency improvements.

We think the time is ripe to seriously consider DNA-based storage and explore system designs and architectural implications. Our ASPLOS paper was the first to address two

Published by the IEEE Computer Society

fundamental challenges in building a viable DNA-based storage system. First, how should such a storage medium be organized? We demonstrate the tradeoffs between density, reliability, and performance by envisioning DNA storage as a key-value store. Multiple key-value pairs are stored in the same pool, and multiple such pools are physically arranged into a library. Second, how can data be recovered efficiently from a DNA storage system? We show for the first time that random access to DNA-based storage pools is feasible by using a polymerase chain reaction (PCR) to amplify selected molecules for sequencing. Our wet lab experiments validate our approach and point to the long-term viability of DNA as an archival storage medium.

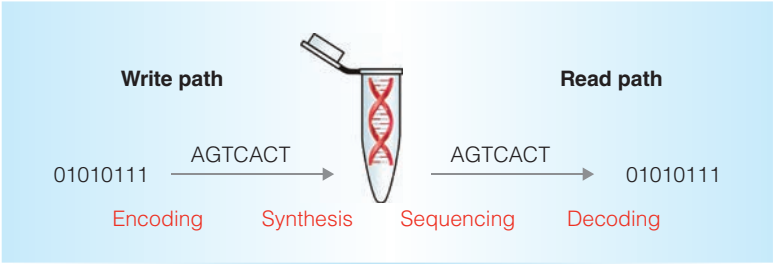

Figure 1. Using DNA for digital data storage. Writes to DNA first encode digital data as nucleotide sequences and then synthesize (manufacture) molecules. Reads from DNA first sequence (read) the molecules and then decode back to digital data.

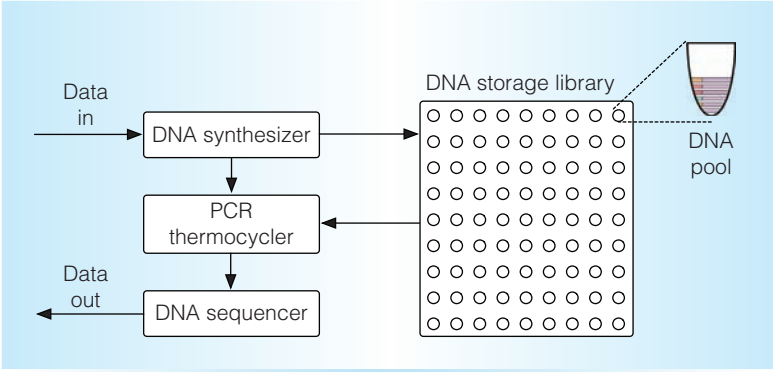

Figure 2. Overview of a DNA storage system. Stored molecules are arranged in a library of pools.

System Design

A DNA storage system (see Figure 2) takes data as input, synthesizes DNA molecules to represent that data, and stores them in a library of pools. To read data back, the system selects molecules from the pool, amplifies them with PCR (a standard process from biotechnology), and sequences them back to digital data. We model the DNA storage system as a key-value store, in which input data is associated with a key, and read operations identify the key they wish to recover.

Writing to DNA storage involves encoding binary data as DNA nucleotides and synthesizing the corresponding molecules. This process involves two non-trivial steps. First, although there are four DNA nucleotides (A, C, G, T) and so a conversion from binary appears trivial, we instead convert binary data to base 3 and employ a rotating encoding from ternary digits to nucleotides.[7] This encoding avoids homopolymers—repetitions of the same nucleotide—that significantly increase the chance of errors.

Second, DNA synthesis technology effectively manufactures molecules one nucleotide at a time, and cannot synthesize molecules of arbitrary length without error. A reasonably efficient strand length for DNA synthesis is 120 to 150 nucleotides, which gives a maximum of 237 bits of data in a single molecule using this ternary encoding. The write process therefore fragments input data into small blocks that correspond to separate DNA molecules. This blocking approach also enables added redundancy. Previous work overlapped multiple small blocks,[7] but our experimental and simulation results show this approach to sacrifice too much density for little gain. Our ASPLOS experiments instead used an XOR encoding, in which each consecutive pair of blocks is XORed together to form a third redundancy block. Although this encoding is simple, we showed that it achieves similar redundancy properties to existing approaches with much less density overhead. Since publishing this paper, our team has been exploring more sophisticated encodings, such as Reed-Solomon codes.

Random Access

Reading from DNA storage involves sequencing molecules and decoding their data back to binary (using the inverse of the encoding discussed earlier). In existing work on DNA storage, recovering data meant sequencing all

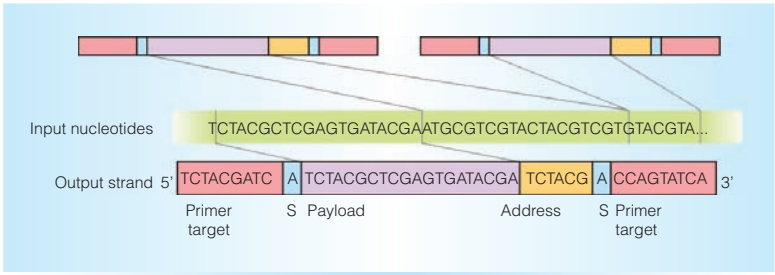

Figure 3. Layout of individual DNA strands. Each strand must carry an explicit copy of its address, because DNA molecules do not offer the spatial organization of traditional storage media.

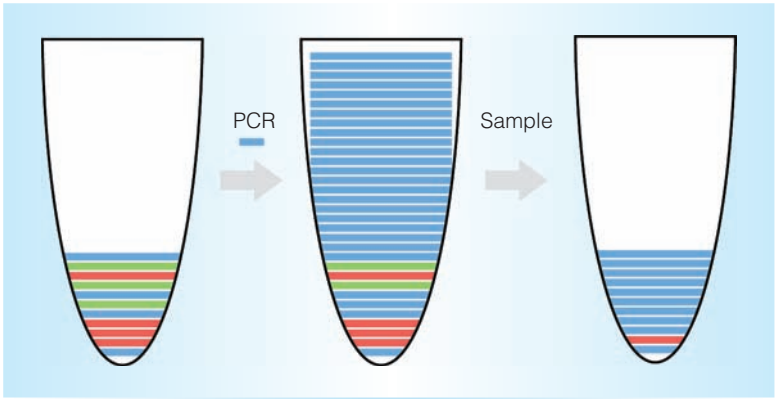

Figure 4. Polymerase chain reaction (PCR) amplifies selected strands to provide efficient random access. The resulting pool after sampling contains primarily the strands of interest.

synthesized molecules and decoding all data at once. However, a realistic storage system must offer random access—the ability to select individual files for reading—if it is to be practical at large capacities.

Because DNA molecules do not offer spatial organization like traditional storage media, we must explicitly include addressing information in the synthesized molecules. Figure 3 shows the layout of an individual DNA strand in our system. Each strand contains a payload, which is a substring of the input data to encode. An address includes both a key identifier and an index into the input data (to allow data longer than one strand). At each end of the strand, special primer sequences—which correspond to the key identifier—allow for efficient sequencing during read operations. Finally, two sense nucleotides ("S") help determine the direction and complementarity of the strand during sequencing.

Our design allows for random access by using PCR, shown in Figure 4. The read process first determines the primers for the given key (analogous to a hash function) and synthesizes them into new DNA molecules. Then, rather than applying sequencing to the entire pool of stored molecules, we first apply PCR to the pool using these primers. PCR amplifies the strands in the pool whose primers match the given ones, creating many copies of those strands. To recover the file, we now take a sample of the product pool, which contains a large number of copies of all the relevant strands but only a few other irrelevant strands. Sequencing this sample therefore returns the data for the relevant key rather than all data in the system.

Although PCR-based random access is a viable implementation, we don't believe it is practical to put all data in a single pool. We instead envision a library of pools offering spatial isolation. We estimate each pool to contain about 100 Tbytes of data. An address then maps to both a pool location and a PCR primer. Figure 5 shows how the random access described earlier fits in a system with a library of DNA pools. This design is analogous to a magnetic-tape storage library, in which robotic arms are used to retrieve tapes. In our proposed DNA-based storage system, DNA pools could be manipulated and necessary reactions could be automated by fluidics systems.

Wet Lab Experiments

To demonstrate the feasibility of DNA storage with random access, we encoded and had DNA molecules synthesized for four image files totaling 151 Kbytes. We then selectively recovered 42 Kbytes of this image data using our random access scheme. We used both an existing encoding[7] and our XOR encoding. We were able to recover files encoded with XOR with no errors. Using the previously existing encoding resulted in a 1-byte error. In total, the encoded files required 16,994 DNA strands, and sequencing produced a total of 20.8 million reads of those strands (with an average of 1,223 reads per DNA strand, or *depth* of 1,223).

To explore the impact of lower sequencing depth on our results, we performed an

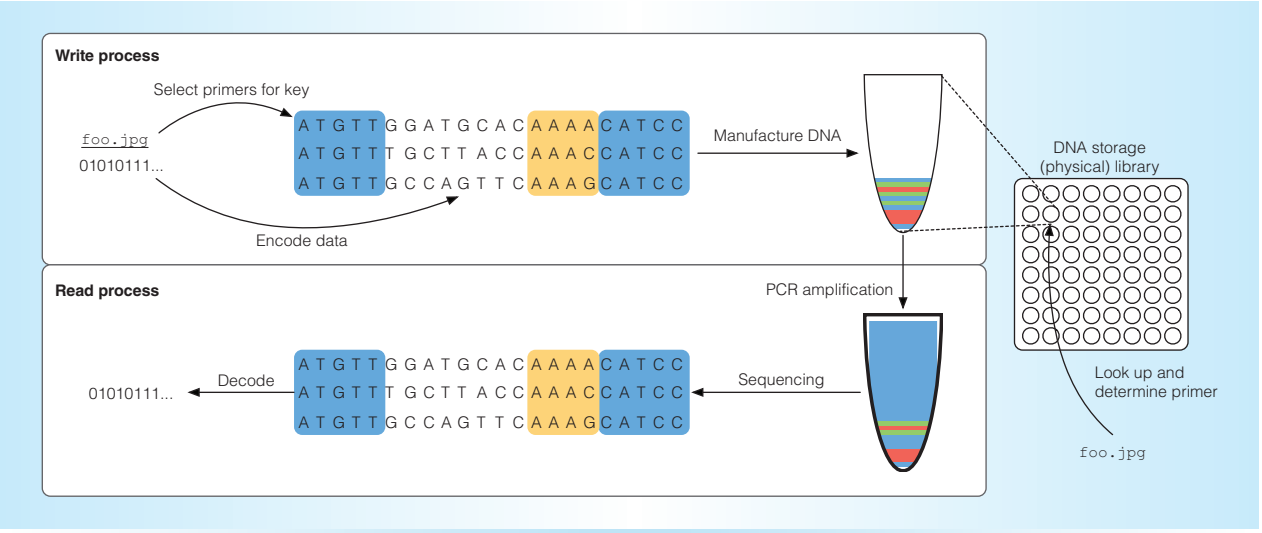

Figure 5. Putting it all together: random access with a pool library for physical isolation. The key data (here, `foo.jpg`) is used with a hash function to identify the relevant pool within the library.

experiment in which we discarded much of the sequencing data (see Figure 6). Lower depth per DNA sequence frees up additional sequencing bandwidth for other DNA sequences, but could omit some strands entirely if they are not sequenced at all. Despite such omissions, the results show that we can successfully recover all data using as few as 1 percent of the sequencing results, indicating we could have recovered 100 times more data with the same sequencing technology. Future sequencing technology is likely to continue increasing this amount.

To inform our coding-scheme design, we assessed errors in DNA synthesis and sequencing by comparing the sequencing output of two sets of DNA sequences with the original reference data. The first set includes the sequences we used to encode data, which were synthesized for our storage experiments by a supplier using an array method. Errors in these sequencing results could be caused either by sequencing or synthesis (or both). The second set includes DNA that was synthesized by a different supplier using a process that's much more accurate (virtually no errors), but also much costlier. Errors in these sequencing results are essentially caused only by the sequencing process. By comparing the two sets of results, we can determine the error rate of both sequencing (results from the second set) and array synthesis (the difference between the two sets). Our results indicate that overall errors per base are a little more than 1 percent and that sequencing accounts for most of the error (see Figure 7).

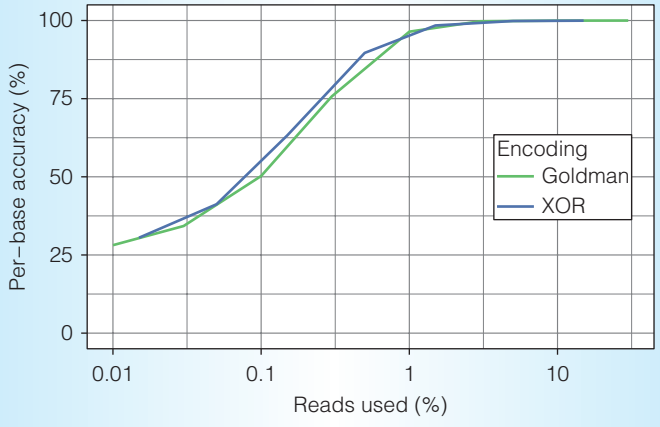

Figure 6. Decoding accuracy as a function of sequencing depth. We successfully recover all data using as little as 1 percent of the sequencing results, suggesting current sequencing technology can recover up to 100 times more data.

Technology Trends

With demand for storage growing faster than even optimistic projections of current technologies, it is important to develop new sustainable storage solutions. A significant

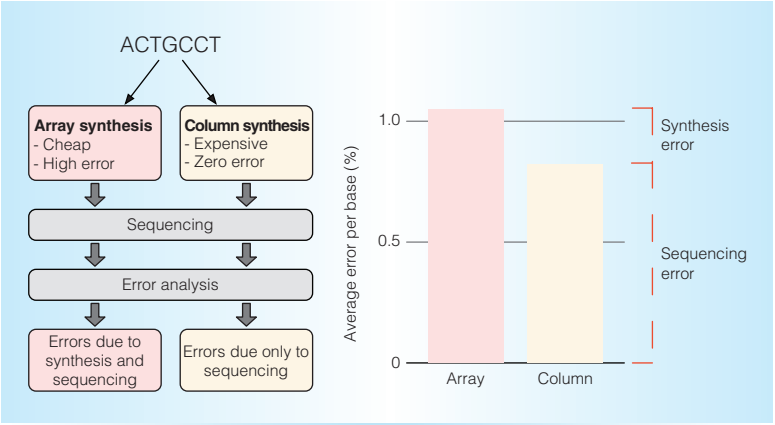

Figure 7. Analysis of error from synthesis and sequencing. Overall errors per base are little more than 1 percent and are mostly attributable to sequencing.

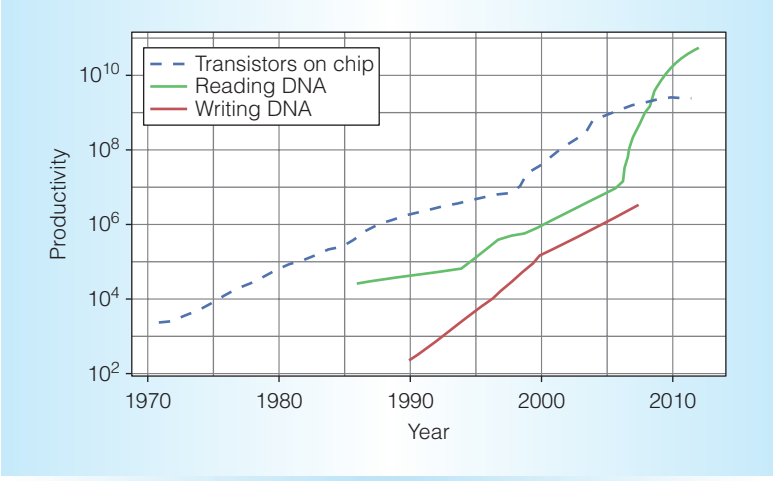

Figure 8. Carlson curves compare trends in DNA synthesis and sequencing to Moore's law.[12] Recent growth in sequencing technology outpaces Moore's law. (Data provided by Robert Carlson.)

fraction of the world's data can be stored in archival form. For archival purposes, as long as there is enough bandwidth to write and read data, latency can be high, as is the case for DNA data storage systems.

Archival storage should be dense to occupy as little space as possible, be very durable to avoid continuous rewriting operations, and have low power consumption at rest because it is meant to be kept for long periods of time. DNA fulfills all these criteria, because it is ultra-dense (1 exabyte per cubic inch for a practical system), is very durable (millennia scale), and has low power requirements (keep it dark, dry, and slightly cooler than room temperature). As we showed in our work, DNA can also support random access, allowing most data to remain at rest until needed.

Current DNA technologies do not yet offer the throughput necessary to support a practical system—in our experiments, throughput was on the order of kilobytes per week. But a key reason for choosing DNA as storage media, rather than some other biomolecule, is that there is already significant momentum behind improvements to DNA manipulation technology. The Carlson curves in Figure 8 compare progress in DNA manipulation technology (both sequencing and synthesis) to improvements in transistor density.[12] Sequencing continues to keep up with, and sometimes outpace, Moore's law. New technologies such as nanopore sequencing promise to continue this rate of improvement in the future.[13]

Future Directions

Using DNA for data storage opens many research opportunities. In the short term, because DNA manipulation is relatively noisy, it requires coding-theoretic techniques to offer reliable behavior with unreliable components. Our team has been working on adopting more sophisticated encoding schemes and better calibrating them to the stochastic behavior of molecular storage. DNA storage also involves much higher access latency than digital storage media, suggesting new research opportunities in latency hiding and caching. Finally, the compactness of DNA-based storage, together with the necessity for wet access to molecules, could open new datacenter-level organizations and automation opportunities for biological manipulation.

In the long term, a last layer of the storage hierarchy with unprecedented density and durability opens up the possibility of storing all kinds of records for extended periods of time. Figure 9 illustrates a possible hierarchy with the properties of each layer. Data that could be preserved for a long time include both system records, such as search and security logs, as well as human records, such as health and historical data in textual, audio, and video formats. Besides its obvious uses in disaster recovery, this opportunity could one

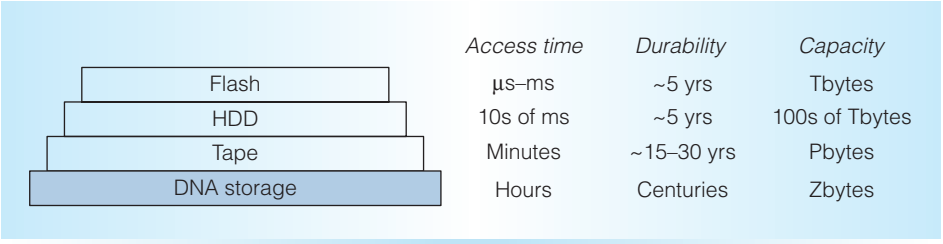

Figure 9. A possible storage system hierarchy. DNA storage is a promising new bottom layer, offering higher density and durability at the cost of latency.

day be a great contributor to the field of digital archeology, the study of human history through "ancient" digital data.

The success of the initial project, published in our ASPLOS paper, motivated us to significantly expand our efforts to explore DNA-based data storage. We formed the Molecular Information Systems Lab (MISL), with members from the University of Washington and Microsoft Research. MISL has worked with Twist Bioscience to synthesize a 200-Mbyte DNA pool,[11] more than three orders of magnitude larger than our ASPLOS results, and an order of magnitude larger than the prior state of the art.[14] Some of its more recent efforts include new coding schemes, sequencing with nanopore-based techniques, and fluidics automation.

Given the impending limits of silicon technology, we believe that hybrid silicon and biochemical systems are worth serious consideration. Now is the time for architects to consider incorporating biomolecules as an integral part of computer design. DNA-based storage is one clear, practical example of this direction. Biotechnology has benefited tremendously from progress in silicon technology developed by the computer industry; perhaps now is the time for the computer industry to borrow back from the biotechnology industry to advance the state of the art in computer systems. MICRO

Acknowledgments

We thank the members of the Molecular Information Systems Laboratory for their continuing support of this work. We thank Brandon Holt, Emina Torlak, Xi Wang, the Sampa group at the University of Washington, and the anonymous reviewers for feedback on this work. This material is based on work supported by the National Science Foundation under grant numbers 1064497 and 1409831, by gifts from Microsoft Research, and by the David Notkin Endowed Graduate Fellowship.

References

1. "Where in the World Is Storage: Byte Density Across the Globe," IDC, 2013; www.idc.com/downloads/where_is_storage_infographic_243338.pdf.
2. "Sony Develops Magnetic Tape Technology with the World's Highest Recording Density," press release, Sony, 30 Apr. 2014; www.sony.net/SonyInfo/News/Press/201404/14-044E.
3. J. Plafke, "New Optical Laser Can Increase DVD Storage Up to One Petabyte," blog, 20 June 2013; www.extremetech.com/computing/159245-new-optical-laser-can-increase-dvd-storage-up-to-one-petabyte.
4. R. Miller, "Facebook Builds Exabyte Data Centers for Cold Storage," blog, 18 Jan. 2013; www.datacenterknowledge.com/archives/2013/01/18/facebook-builds-new-data-centers-for-cold-storage.
5. G.M. Church, Y. Gao, and S. Kosuri, "Next-Generation Digital Information Storage in DNA," *Science*, vol. 337, no. 6102, 2012, pp. 1628–1629.
6. C.T. Clelland, V. Risca, and C. Bancroft, "Hiding Messages in DNA Microdots," *Nature*, vol. 399, 1999, pp. 533–534.
7. N. Goldman et al., "Towards Practical, High-Capacity, Low-Maintenance Information Storage in Synthesized DNA," *Nature*, vol. 494, 2013, pp. 77–80.
8. M.E. Allentoft et al., "The Half-Life of DNA in Bone: Measuring Decay Kinetics in 158

Dated Fossils," *Proc. Royal Society of London B: Biological Sciences*, vol. 279, no. 1748, 2012, pp. 4724–4733.

9. J. Bornholt et al., "A DNA-Based Archival Storage System," *Proc. 21st Int'l Conf. Architectural Support for Programming Languages and Operating Systems* (ASPLOS), 2016, pp. 637–649.
10. M. Brunker, "Microsoft and University of Washington Researchers Set Record for DNA Storage," blog, 7 July 2016; http://blogs.microsoft.com/next/2016/07/07/microsoft-university-washington-researchers-set-record-dna-storage.
11. L Organick et al., "Scaling Up DNA Data Storage and Random Access Retrieval," bioRxiv, 2017; doi:10.1101/114553.
12. R. Carlson, "Time for New DNA Synthesis and Sequencing Cost Curves," blog, 12 Feb. 2014; www.synthesis.cc/2014/02/time-for-new-cost-curves-2014.html.
13. "Oxford Nanopore Technologies," http://nanoporetech.com.
14. M. Blawata et al., "Forward Error Correction for DNA Data Storage," *Procedia Computer Science*, vol. 80, 2016, pp. 1011–1022.

James Bornholt is a PhD student in the Paul G. Allen School of Computer Science and Engineering at the University of Washington. His research interests include programming languages and formal methods, focusing on program synthesis. Bornholt received an MS in computer science from the University of Washington. Contact him at bornholt@cs.washington.edu.

Randolph Lopez is a graduate student in bioengineering at the University of Washington. His research interests include the intersection of synthetic biology, DNA nanotechnology, and molecular diagnostics. Lopez received a BS in bioengineering from the University of California, San Diego. Contact him at rmlb@uw.edu.

Douglas M. Carmean is a partner architect at Microsoft. His research interests include new architectures on future device technology. Carmean received a BS in electrical and electronics engineering from Oregon State University. Contact him at dcarmean@microsoft.com.

Luis Ceze is the Torode Family Associate Professor in the Paul G. Allen School of Computer Science and Engineering at the University of Washington. His research interests include the intersection between computer architecture, programming languages, and biology. Ceze received a PhD in computer science from the University of Illinois at Urbana–Champaign. Contact him at luisceze@cs.washington.edu.

Georg Seelig is an associate professor in the Department of Electrical Engineering and the Paul G. Allen School of Computer Science and Engineering at the University of Washington. His research interests include understanding how biological organisms process information using complex biochemical networks and how such networks can be engineered to program cellular behavior. Seelig received a PhD in physics from the University of Geneva. Contact him at gseelig@uw.edu.

Karin Strauss is a senior researcher at Microsoft Research and an affiliate faculty at the University of Washington. Her research interests include studying the application of biological mechanisms and other emerging technologies to storage and computation, and building systems that are efficient and reliable with them. Strauss received a PhD in computer science from the University of Illinois at Urbana–Champaign. Contact her at kstrauss@microsoft.com.