Universal Library of Innovative Research and Studies Research Article

ISSN: 3065-0003 | Volume 2, Issue 4

Open Access | PP: 37-42

DOI: https://doi.org/10.70315/uloap.ulirs.2025.0204006

Improving Scientific Computing Efficiency in Pharma Using Cost-Optimized Cloud Infrastructure

Nivedha Sampath

Platform Engineer at Takeda Pharmaceuticals, Boston, USA.

Abstract

The article considers the pharmaceutical industry's transition to cost-optimized cloud infrastructure for scientific computing as a strategic transformation of research processes. The relevance of the work is driven by the exponential growth in data volumes in molecular dynamics, genomics, and proteomics, as well as the need for accelerated training of large artificial intelligence models, which makes traditional enterprise clusters economically and technically inefficient. The study was framed as an effort to seek the models of consumption of cloud resources and hardware profiles that would lower costs while maintaining or improving computing performance for pharmaceutical science. The novelty in the article is in its absolute scrutiny of hybrid consumption models (spot, reserved, and serverless instances) to be used in combination with accelerators specialized in GPUs, TPUs, FPGAs, and quantum processors on one side and a systematized view of FinOps practices tying research tasks together with transparent cost allocation on the other. The main conclusion about efficiency is based on two factors: first, a very exact match between the hardware stack and workload profile; second, tight organization within the resource consumption model. Pharmaceutical companies can increase the speed of research without growth in total expenditures by justified use of next-generation GPU series and energy-efficient Arm hosts, plus specialized accelerators, combined with a distributed data storage inclusive network-flow control. Organizationally, the key factor is the introduction of continuous cost-optimization practices (FinOps) and iterative infrastructure scaling through pilot projects and declarative manifests. The article will be helpful to researchers in pharmaceutical science, cloud-platform engineers, and R&D leaders responsible for digitization strategy and reducing the costs of computing processes.

Keywords: *Pharmaceutical Computing, Cloud Infrastructure, HPC, FinOps, Molecular Dynamics, Genomics, Proteomics, Artificial Intelligence, Cost Optimization.*

INTRODUCTION

Computer simulation has now assumed a main role in pharmaceutical research: molecular-dynamics paths, "omics" checks, and teaching big models all need more and more high-level computing. Thus, the high-performance computing (HPC) field for life sciences is seeing a compound yearly growth rate of 11.6% up to 2031 (Insightace Analytic Pvt Ltd, 2025). A like jump is noted in side metrics: the MDverse project has listed over 250,000 files from 2,000 scientific data sets placed by experts in open stores. It emphasizes that the rate of posting such simulations continues to grow (Tiemann et al., 2024).

However, the enterprise clusters on which the industry has traditionally relied have rigid structural constraints. Before the widespread adoption of virtualization, the average utilization of physical servers fluctuated at only 12–18%, meaning most racks consumed power while contributing little to computation (Energy Star, n.d.). Every new project meant weeks of buying equipment and millions spent on capital expenditures plus costly support for infrastructure;; it is not a coincidence, as VentureBeat analysts put it, that the shift to cloud reduces direct infrastructure costs by an average of 43% (Keefe, 2025).

Workloads are being shifted to an elastic cloud, and instance-selection policies and purchasing modes are being retuned so that each mission gets exactly the resources it requires, not a cent more. Already about 83% of pharmaceutical firms use the cloud for some operations in hybrid mode while gradually refactoring their legacy pipelines, according to HIMSS (Keefe, 2025). The strategic plan is an implementation strategy designed purposely to transform fixed capital expenditures into flexible operating ones while providing new molecules' time-to-market acceleration and freeing up direct budget for scientific experimentation.

Citation: Nivedha Sampath, "Improving Scientific Computing Efficiency in Pharma Using Cost-Optimized Cloud Infrastructure", Universal Library of Innovative Research and Studies, 2025; 2(4): 37-42. DOI: https://doi.org/10.70315/uloap.ulirs.2025.0204006.

MATERIALS AND METHODOLOGY

The study was based on an analysis of academic publications, industry reports, technical documentation from cloud providers, and practical case studies on deploying HPC in the pharmaceutical industry. The foundation included sources reflecting both the growth dynamics of the HPC market for life sciences (Insightace Analytic Pvt Ltd, 2025) and the practice of open scientific communities, where mass sharing of molecular-dynamics trajectories has confirmed a trend toward exponential data increase (Tiemann et al., 2024). Additional materials were considered on the energy efficiency of traditional data centers (Energy Star, n.d.), as well as analytics on comparative costs when moving to the cloud (Keefe, 2025). This corpus made it possible to cover both academic and applied perspectives.

Methodologically, the work combined three research approaches. First, a comparative analysis was conducted of hardware profiles and cloud resource-consumption models, including cases of scalable GROMACS simulations and hybrid quantum-classical pipelines, which made it possible to assess not only computational efficiency but also pricing scenarios when choosing GPUs, TPUs, FPGAs, and specialized accelerators (Kutzner et al., 2022; Zhao et al., 2025). Second, a systematic review was performed of regulatory and market reports, including survey data on the extent of cloud adoption by pharmaceutical companies, thereby aligning the macroeconomic context with corporate strategies (Keefe, 2025). Third, a content analysis was applied to practical cases from genomics and proteomics, which illuminated issues of scale and ways to solve them through cloud SaaS architectures and distributed storage (Illumina, n.d.; Li et al., 2024).

RESULTS AND DISCUSSION

The industry's primary computational workloads fall into three complementary families. The first comprises molecular dynamics, virtual docking, and high-precision quantum-chemical estimation of reaction energies. In a recent GROMACS benchmark in the cloud, it was possible to simultaneously employ more than 4,000 instances, 140,000 CPU cores, and 3,000 GPUs, reducing a cycle of 19,872 simulations from weeks to two days without increasing total expenditures when spot-instance selection and checkpoint protocols were configured correctly (Kutzner et al., 2022). For more complex electronic correlations, hybrid quantum-classical pipelines already demonstrate triple-digit time savings: IonQ, together with AstraZeneca and AWS, accelerated modeling of a key step of the Suzuki–Miyaura reaction by more than twentyfold, moving the task from "months" to "days" and confirming that quantum accelerators are beginning to justify their class of resources specifically in drug-discovery scenarios (Zhao et al., 2025).

The second family is associated with processing genomic and proteomic data. A single NovaSeq 6000 installation outputs up to six terabases of data and twenty billion reads in less than forty-eight hours, instantly turning local storage into a bottleneck and forcing alignment and variant analysis into S3 or Google Cloud Storage objects for subsequent DRAGEN- or Spark-based processing (Illumina, n.d.). In proteomics, the increase in mass-spectrometric sensitivity produces a similar effect: the CloudProteoAnalyzer service distributed peptide identification across many nodes of a supercomputer and showed that a SaaS architecture in the cloud scales without loss of accuracy, leaving local stations primarily for rapid validation of results (Li et al., 2024). Its architecture is shown in Figure 1.

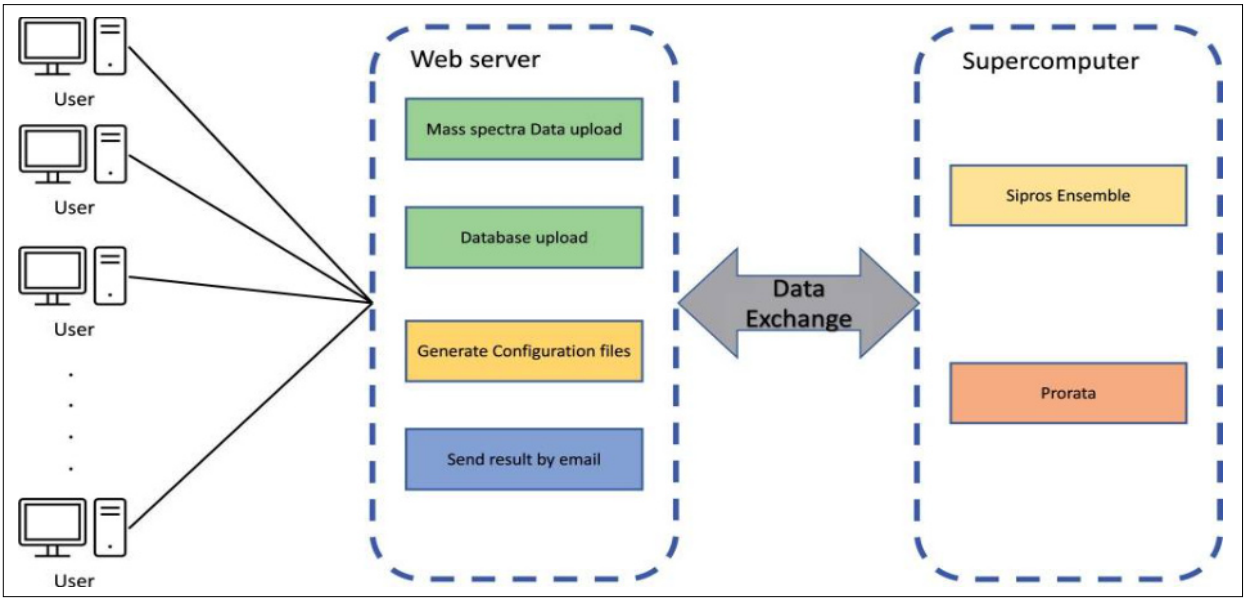

Fig. 1. CloudProteoAnalyzer architecture (Li et al., 2024)

The third direction is training and inference of artificial intelligence models for predicting molecular structures and properties. The classic AlphaFold pipeline (Figure 2) required more than eleven days of pretraining on 128 TPUs, i.e., approximately 34,000 TPU-hours (Zhu et al., 2024).

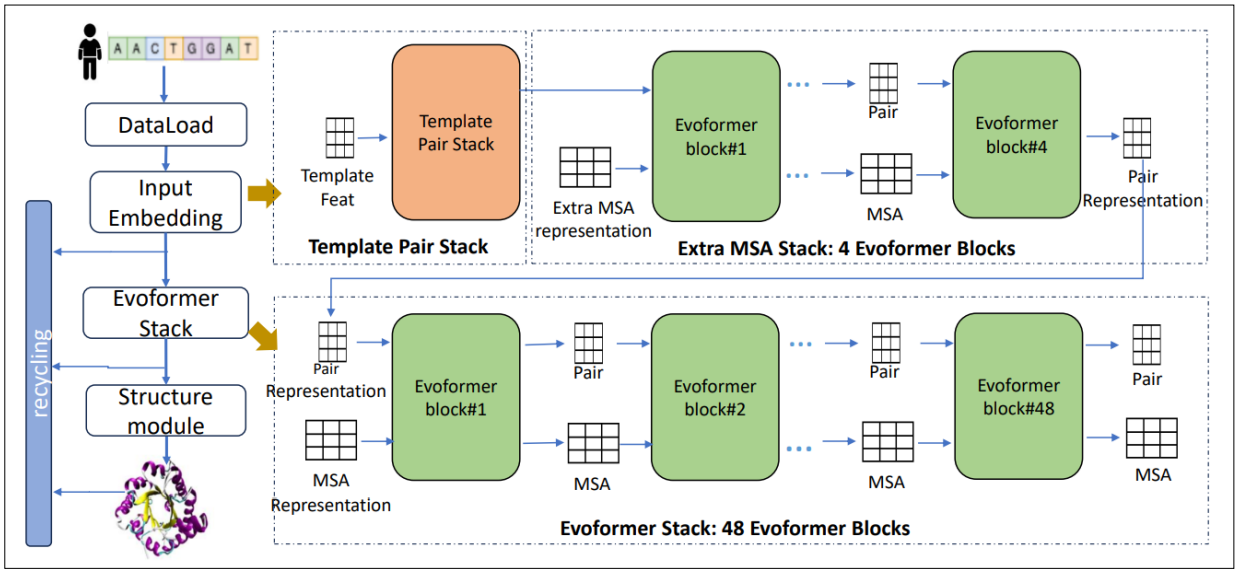

Fig. 2. Structure of the AlphaFold model (Zhu et al., 2024)

The ScaleFold methodology optimized communications and kernels, stretching parallelism to 2,080 NVIDIA H100s and reducing the same stage to ten hours, which is equivalent to almost an eightfold acceleration and a direct saving of the budget for the cloud GPU pool (Zhu et al., 2024). In inference, the same models once quantized and deployed on Arm servers manage batch QSAR-screening requests with latency of less than one hundred milliseconds, hence intermediate metadata are moved to object storage and cached only for the time of the active virtual-screening cycle.

Altogether, these three workload types are the reason "compute-as-needed" hybrid scheduler architectures that can switch among GPUs, CPUs, spot pricing, and quantum accelerators are becoming the technical standard in the scientific divisions of pharmaceutical companies.

Cloud infrastructure is best determined by how well the hardware stack matches the workload profile. For heavy HPC compute and large-model training, it is recommended to use the newest GPU instances with H100 chips: From Amazon's public documentation, the new P5 series shortens training time fourfold and at the same time reduces total expenditures for deep learning and HPC by about forty percent compared with the previous generation (Amazon Web Services, Inc., n.d.). Such accelerators scale well in clusters on the order of tens of thousands of GPUs and thus remain the baseline choice for pharmaceutical simulations and generative structure models.

A whole belt of service components operates around heavy compute: orchestration systems, ETL pipelines, and metadata stores. They rarely require high-performance cores, yet are cost-sensitive. Moving these tasks to Graviton Arm servers can save up to forty percent on a price-performance basis relative to comparable x86 nodes due to better energy consumption and the absence of hyper-threading overhead (Hykell, n.d.). The pivot is made with small code changes, so firms use Arm as a cost-effective layer for supporting microservices and off-peak reporting. For tight algorithms where matrix-work rate or ultra-low lag is key, the cloud has special boosters. Google's new Trillium tensor chips show nearly a five times speed boost and a sixty-seven percent energy cut versus TPU v5e, making them suitable for big-scale guessing of drug models on set budgets (Cherney, 2024). In genomics, FPGAs—for instance, the FAST setup for adapter trimming—cut the time of matching steps by a tenfold without losing accuracy, thus moving the main variant-check phase out of the central path (Khaleghi et al., 2022). Mixing such special chips with general-use GPUs and Arm hosts gives the best price-speed mix at every step in the science cycle.

Cost optimization in the cloud begins with choosing the resource-consumption model, because even the most modern hardware stack loses efficiency if paid at an inflated on-demand rate. For compute stages that can be safely interrupted after regular checkpoints, spot instances remain the best option: according to the AWS document "Compute on AWS: How to Choose," unused capacity is sold at discounts of up to 90% off the regular price, with interruptions accompanied by two minutes of advance notice, which allows R&D teams to move large pools of molecular dynamics, docking, or preliminary molecule triage to spot without risk (Amazon Web Services, Inc., 2025). Current HPC research, such as the Uniform Progress algorithm, further shows that smart redistribution of tasks between spot and reserve yields 27–84% savings while meeting agreed deadlines, which is critical for cycles tied to the end dates of clinical phases (Wu et al., 2024).

When a project is tied to a rigid schedule for publication of results or a registration dossier, disruptions are unacceptable, and capacity is reserved in advance. In effect, pharma apps run a hybrid portfolio: place the critical jars on reserved instances and run limbs at spot, achieving both reliability and price advantages. For ephemeral microservices like format conversion, integrity verification, or visualization scripts,

the best practice is to use the Function-as-a-Service model. MDPI study over sustainable IT deployment informs, moving to serverless cuts operational expenditures up to 60% as payment is only for function execution time and auto scaling on real traffic peak (Akour & Alenezi, 2025). Within scientific pipelines, this is especially noticeable in data-preparation stages of multi-omics analysis, where each event lasts mere seconds but triggers thousands of parallel processes. At the same time, the global artificial intelligence in drug discovery market size was estimated at USD 1.5 billion in 2023 and is projected to reach USD 20.30 billion by 2030, growing at a CAGR of 29.7% from 2024 to 2030 (GVR, n.d.).

Thus, a rational combination of spot, reserved, and serverless models turns the cloud from a simple source of capacity into a managed financial instrument: inexpensive spot covers massive parallel computation, reservations guarantee continuity of key branches, and functions handle small but numerous tasks, preserving budget for the most important goal—accelerating the time-to-market of drug molecules.

A good cloud setup comes from clearly splitting the compute part and the storage part, so that the amount of processor power can be changed without touching data banks. This split is very useful in drug-making work, where high compute needs happen in short runs while base and middle files are still needed for a long time. When storage is spread across object services, the cluster heart is not a single point of risk, and the storage budget is freed up for more important jobs.

A multicloud storage strategy permits keeping operational data closer to the compute core and, at the same time, automatically moving historical layers to a colder, more economical region. When tiering policies are keyed on frequency of access, active trajectories and models stay on hot disks within the same cloud that the containers run in; infrequently requested replicas and checkpoints can be placed in more cost-effective storage on another platform. This reduces the total cost of ownership without extra administration and simultaneously creates a backup with an independent provider.

Network costs become a new optimization center, because moving peta- and exabyte-scale datasets between sites can quickly negate the benefits of inexpensive storage. The solution is to localize computing where the most extensive data reside, use on-the-fly compression and aggregated transfer windows, and carefully control traffic directions. Correct routing decreases intercloud egress plus increases real throughput at the same provisioned bandwidth cap, making data get quicker for the next stage of research. Rational apportionment of cloud expenses requires as much scientific rigor as experiments in computing itself. Once data and processes have been distributed over several sites, all billing events will be coming into a unified log stream that gives each virtual machine, container, or function a tag with the project identifier and the pipeline stage. End-to-end attribution shows precisely how much comparative modeling or activity checking of molecules costs, turning financial indicators into another metadata layer that researchers can analyze alongside simulation results.

When transparency is achieved, chargeback and showback models come into play. The first debits are costs directly to laboratories or programs, forcing teams to account for budget in planning replica counts and restart frequency. The second keeps payment on a central account but regularly publishes details, fostering healthy competition among divisions for resource-use efficiency. In both cases, cost visibility leads to voluntary abandonment of irrational pools and encourages migration to cheaper instance classes in stages where performance is not the limiting factor.

The FinOps loop is closed by continuous optimization based on hardware-generation refresh. As soon as a new processor or accelerator family appears in the cloud, an automated testbed runs standard tasks, comparing execution time and hourly price. If the total cost is lower at comparable accuracy, the infrastructure code migrates to the newer architecture, and the old family is gradually removed from deployment templates. This cycle of constant configuration review guarantees that the economic effect does not erode over time and that the budget maximizes the scientific value of computations.

Implementation of an optimized cloud platform begins with a detailed survey of existing workflows and the actual profile of computational needs. All stages, from the generation of source data to the publication of results, are mapped by engineers and research representatives in joint efforts. This includes recording the time spent on each task, its type, average duration, memory requirements, and storage specifics. A heat map of expenditures is based on these observations. Infrastructure bottlenecks and excess reserves are immediately visible. Besides, it provides insight into which part of the pipeline will yield the savings if moved to the cloud.

A minimally viable pilot is then constituted. One representative project is selected, and it provides at least one example of each key workload: simulation tasks, processing large biological datasets, and machine-learning models. The pilot gets started on a small number of compute instances running under predefined metrics of execution time, resource cost per unit of scientific output, and reproducibility stability. If the indicators improve relative to the baseline, the results are recorded in a report that separately outlines recommendations for applying the same principles to adjacent processes.

After validating the pilot phase, stepwise scaling begins. The new infrastructure template has been added to version control, and deployment processes have been moved from manual mode to declarative manifests, allowing each laboratory to set up an independent environment as needed. Meanwhile, a series of practical workshops is available for users to learn how to initiate tasks through the orchestrator, tag usage for expenditure tracking, and review FinOps reports. The architecture is illustrated in Figure 3.

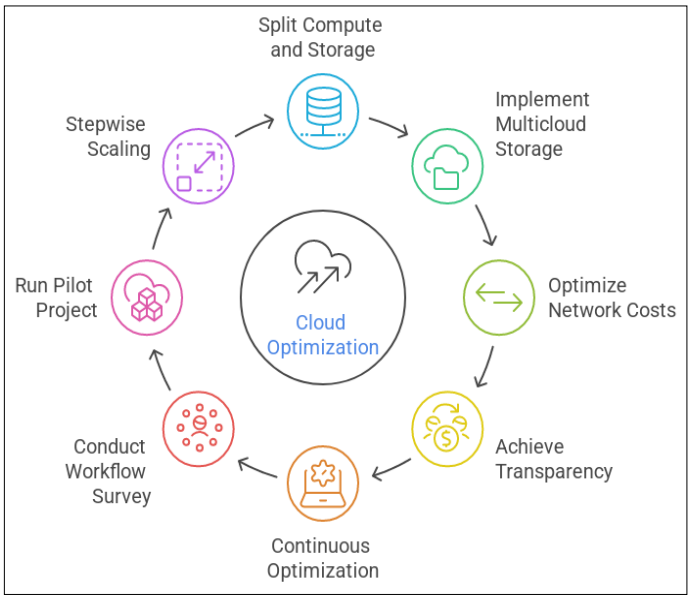

Fig. 3. Cloud Optimization Cycle

An ongoing improvement loop captures participant feedback to help steady the retuning of the architecture as volumes of data accumulate and new experiment types emerge, while remaining cost-transparent and highly flexible to meet scientific needs.

CONCLUSION

The article demonstrates that the transition of pharmaceutical science to optimized cloud infrastructure is less a matter of infrastructure savings than a systemic transformation of the computing cycle, aimed at accelerating scientific iterations and converting capital expenditures into operating expenditures. The examples cover three complementary classes of workloads—molecular dynamics and quantum-chemical calculations, processing of "omics" data, and training of large models—and demonstrate that, with a well-chosen hardware stack and resource-consumption model, the cloud can substantially reduce computation time without increasing total expenditures. Savings and acceleration illustrations comprise scalable GROMACS benchmarks, hybrid quantum-classical pipeline examples, and optimized training for models like AlphaFold, which underscore the pragmatic advantages of merging specialized accelerators, scaling, and checkpoint approaches.

The scientific computing performance depends, technically speaking, not on the strength of individual cores but on how well and to what acceptable degree the entire hardware stack fits the workload profile. Therefore, Modern GPU series are justified for heavy HPC tasks and training large models. Workloads for such ancillary services and ETL should be better moved closer to energy-efficient Arm hosts. Different stages require looking toward tensor accelerators, FPGAs, and other special chips. The separation of the compute and storage layers, together with a multicloud tiering policy, would own the data while reducing single point of failure risk; Compute localization, plus controlling network flows, keeps big dataset moves under control.

Economic and organizational synthesis reads thus: systematic consumption of resources is what truly brings about cost optimization within the cloud. This involves spot, reserved, and serverless models acting as a managed financial instrument- where cheap spot instances cover extensive parallel computation, reservations provide insurance for the critical branch to be running, and functions for many short-lived tasks. Continuous FinOps practice with resource tagging, end-to-end cost attribution, together with chargeback or showback models, brings transparency whereby expenditure becomes a measurable parameter of research efficiency that optimizes it by the laboratories.

Begin with a detailed workflow survey, build a consumption heat map, pilot minimally viable validate metrics of time and cost per unit of scientific output, and reproducibility, then scale stepwise move deployments to declarative manifests, and organize training for researchers. It is an iteration that will sustain the economic effect over time as new generations of hardware are tested in automated testbeds, reducing template obsolescence risk.

The right hardware profile, joined with flexible consumption models and strong cost-accounting practices, lets pharma firms boost the efficiency of scientific computing. This makes room for a genuine chance to shift freed funds to lab work and speed up bringing new molecules to market while keeping the same level of reliability and reproducibility in research pipelines.

REFERENCES

1. Akour, M., & Alenezi, M. (2025). Reducing Environmental Impact with Sustainable Serverless Computing. *Sustainability*, *17*(7), 2999. https://doi.org/10.3390/su17072999
2. Amazon Web Services, Inc. (n.d.). *Amazon EC2 P5 Instances*. Amazon Web Services, Inc. Retrieved July 29, 2025, from https://aws.amazon.com/en/ec2/instance-types/p5/
3. Amazon Web Services, Inc. (2025). *Choosing an AWS compute service*. Amazon Web Services, Inc. https://docs.aws.amazon.com/pdfs/decision-guides/latest/compute-on-aws-how-to-choose/compute-on-aws-how-to-choose.pdf
4. Cherney, M. (2024, May 14). *Google launches Trillium chip, improving AI data center performance fivefold*. Reuters. https://www.reuters.com/technology/google-launches-trillium-chip-improving-ai-data-center-performance-fivefold-2024-05-14/
5. Energy Star. (n.d.). *Virtualize Servers*. Energy Star. Retrieved July 18, 2025, from https://www.energystar.gov/products/data_center_equipment/5-simple-ways-avoid-energy-waste-your-data-center/virtualize-servers
6. GVR. (n.d.). *Artificial Intelligence In Drug Discovery Market Report*. GVR. Retrieved August 10, 2025, from

https://www.grandviewresearch.com/industry-analysis/artificial-intelligence-drug-discovery-market

7. Hykell. (n.d.). *Performance and cost comparison between AWS Graviton and Intel instances*. Hykell. Retrieved July 31, 2025, from https://hykell.com/kb/graviton-instances/cost-comparison-between-graviton-and-intel-instances/
8. Ilumina. (n.d.). *NovaSeq 6000 System*. Ilumina. Retrieved July 26, 2025, from https://www.illumina.com/systems/sequencing-platforms/novaseq.html
9. Insightace Analytic Pvt Ltd. (2025, April 30). *High-Performance Computing for Life Sciences Market Projected to Grow at a Robust CAGR of 11.6% Driven by Healthcare Innovation*. OpenPR; openPR. https://www.openpr.com/news/3994302/high-performance-computing-for-life-sciences-market-projected
10. Keefe, E. (2025, February 6). *Cloud vs. On-Premises in the Pharmaceutical Industry*. Sikich. https://www.sikich.com/insight/cloud-vs-on-premises-in-the-pharmaceutical-industry-which-delivers-a-lower-total-cost-of-ownership/
11. Khaleghi, B., Zhang, T., Shao, N., Akel, A., Curewitz, K., Eno, J., Eilert, S., Moshiri, N., & Rosing, T. (2022). FAST: FPGA-based Acceleration of Genomic Sequence Trimming. *2022 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 510–514. https://doi.org/10.1109/biocas54905.2022.9948621
12. Kutzner, C., Kniep, C., Cherian, A., Nordstrom, L., Grubmüller, H., de Groot, B. L., & Gapsys, V. (2022). GROMACS in the Cloud: A Global Supercomputer to Speed Up Alchemical Drug Design. *Journal of Chemical Information and Modeling*, *62 (7)*, 1691–1711. https://doi.org/10.1021/acs.jcim.2c00044
13. Li, J., Xiong, Y., Feng, S., Pan, C., & Guo, X. (2024). CloudProteoAnalyzer: scalable processing of big data from proteomics using cloud computing. *Bioinformatics Advances*, *4*(1). https://doi.org/10.1093/bioadv/vbae024
14. Tiemann, J. K., Szczuka, M., Bouarroudj, L., Oussaren, M., Garcia, S., Howard, R. J., Delemotte, L., Lindahl, E., Baaden, M., Lindorff-Larsen, K., Chavent, M., & Poulain, P. (2024). MDverse, shedding light on the dark matter of molecular dynamics simulations. *ELife*, *12*(RP90061). https://doi.org/10.7554/elife.90061
15. Wu, Z., Chiang, W.-L., Mao, Z., Yang, Z., Friedman, E., & Shenker, S. (2024). *Can't Be Late: Optimizing Spot Instance Savings under Deadlines Can't Be Late: Optimizing Spot Instance Savings under Deadlines*. https://www.usenix.org/system/files/nsdi24-wu-zhanghao.pdf
16. Zhao, L., Goings, J. J., Aboumrad, W., Arrasmith, A., Calderin, L., Churchill, S., Gabay, D., Harvey-Brown, T., Hiles, M., Kaja, M., Keesan, M., Kulesz, K., Maksymov, A., Maruo, M., Muñoz, M., Nijholt, B., Schiller, R., de Sereville, Yvette, Smidutz, A., & Tripier, F. (2025). *Quantum-Classical Auxiliary Field Quantum Monte Carlo with Matchgate Shadows on Trapped Ion Quantum Computers*. Arxiv. https://arxiv.org/abs/2506.22408
17. Zhu, F., Nowaczynski, A., Li, R., Xin, J., Song, Y., Marcinkiewicz, M., Eryilmaz, S. B., Yang, J., & Andersch, M. (2024). ScaleFold: Reducing AlphaFold Initial Training Time to 10 Hours. *Arxiv, 265*, 1-6 https://doi.org/10.1145/3649329.3657326