Top 10 Questions IT Leaders Should Ask of Data Science Platforms

A data science platform is where all data science work takes place and acts as the system of record for predictive models. While a few leading model-driven businesses have made the data science platform an integral part of their enterprise architecture, most companies are still trying to understand what a data science platform is and how it fits into their architecture. Data science is unlike other technical disciplines, and models are not like software or data. Therefore, a data science platform requires a different type of technology platform. This document provides IT leaders with the top 10 questions to ask of data science platforms to ensure the platform handles the uniqueness of data science work.

1. Where/how is the platform hosted?

An ideal data science platform should work with existing infrastructure. It provides the flexibility to be hosted in the Cloud (e.g. a VPC—a vendor-managed private cloud), on-premise, or perhaps hybrid. Either way, the platform should be based on a single code-base, regardless of where it is hosted. If business requirements call for changes in infrastructure, the ideal platform provides the flexibility to adapt to those changes.

2. How can the platform help me ensure that data scientists use tools and packages (open-source or proprietary) that have been approved?

Data science requires free-from experimentation and access to the latest revolutions in open-source tooling to achieve breakthroughs. However, enterprises need to provide guardrails on experimentation and tools to guard against breaches and protect company IP. So, a data science platform must support various native data science tools (JupyterLab, RStudio, SAS, etc.) through an open and flexible approach, while providing IT teams the capabilities to govern the data science environments and provision pre-approved environments. This approach will remove the data science shadow IT challenge and ensures IT infrastructure is not exposed to unnecessary risks.

3. How does the platform handle the dynamic nature of data science work?

Data scientists' work requires somewhat unpredictable access to different sizes of hardware, including GPUs, when doing intense work like deep learning. Reserving large hardware instances that sit idle is too expensive, so a data science platform should provide elastic access to different types of machines and software packages. These environments should be available with a single-click, removing DevOps tasks from data scientists' daily work. IT teams should be able to control which users have access to which environments, and also have complete visibility into the costs, time, and usage of each of these environments. Ultimately, the platform should provide ability for parallel execution (running multiple experiments in parallel) in resource provisioning.

4. How does the platform handle user security and increasingly complex governance requirements where data scientists have access to highly sensitive data?

An ideal platform for data science should work with existing user security practices such as Single Sign-On (SSO). However, in data science, providing authorization and authentication security isn't enough. Data science is different, and a complete platform also provides an audit trail of all data science work (code, data, packages, environments, comments) for an individual user that ensures reproducibility and auditability of the users' work. Along with this visibility and auditability, IT should have access to a flexible permission model to govern access to models, projects, data, experiments, hardware, and software packages that scales to support growth to hundreds of users.

5. How does the platform help reduce regulatory and operational risks and help future-proof me from upcoming regulatory hurdles?

Keeping a comprehensive and thorough system of record in the data science lifecycle can significantly reduce regulatory and operational risks. An ideal data science platform preserves the entire lifecycle of a model for a system of record. All revisions of a project should be tracked to enable easy retrieval of any experiment for audits, risk governance, and compliance checks.

For example, a model developed to predict insurance policy holder risk may need to be audited and adjusted based on new personal privacy laws. A full model provenance log would enable one to trace back every step of model creation, understand how specific sensitive personal data impacts the model, and how that sensitive data was used in development of the model. Additionally, a data scientist could start from any point in that model creation process to fork off and develop an updated model without starting from scratch, accelerating new model development while reducing compliance risk.

6. Why do existing tools like, Git, JIRA, and Jenkins fail to meet the needs of a data science platform?

Data science is different than software development; models require re-training, are developed in an experimental fashion, and are made using lots of different software tools. There is no need to "retrain" software code, but production models do need to be retrained frequently. A data science platform provides a single and comprehensive system-of-record (SOR) for models, which is much more than keeping track of code versions and issues. Data science assets include code, data, discussion threads, hardware tiers, software package versions, parameters, results, and more. Git and JIRA are not built for an experimental process. Furthermore, data scientists will reject GIt/Jira/Jenkins built systems since they hinder their work instead of accelerating it. A data science platform accelerates model development and deployment, with access to elastic compute, automatic experiment tracking, full reproducibility, model-based collaboration, streamlined model-deployment, and a knowledge base of building blocks to enable rapid model development.

7. What data does the platform provide access to? And how does it handle the data versioning requirements of data science?

A data science platform needs to provide simple, fast, and secure access to ALL types of data including Hadoop, Spark, flat files, and databases. These connections must be encrypted in transit, be able to handle failover, and set up to transfer large amounts of data for model training and experimentation. Data science also involves lots of data manipulation and creation of new "features," which are created based on other data. Since the data and features often change in each experiment, the snapshot of that data needs to be captured and revisioned so that the model and data is auditable, reproducible, and meets compliance requirements in regulated industries.

8. How does the platform enable user-friendly, enterprise-ready model operations (ModelOps)?

Model operations involves deploying models to production and the process of monitoring, re-training, and updating them in production. Model deployment is the process of enabling a model to be used in production (for example, deploying the model as a simple visual (chart, graph), an interactive application, or as an API) so the model can be used for interactive human consumption or machine-based consumptions. An ideal data science platform should allow data scientists to self-serve and directly deploy models in the various different modes, with IT approval and oversight. Once the model is deployed, the platform should monitor model performance, provide ability to retrain, and revision that model in production, capturing full model provenance for audit records. Lastly, the platform should ensure that end-users have a direct feedback path, from the model to the data scientists, to ensure rapid iteration on the model.

9. How does the platform help govern cloud infrastructure costs and plan for future technology needs?

A data science platform should provide an elastic and flexible compute infrastructure to meet the dynamic resource requirements of data science projects. Poor resource provisioning can lead to unexpectedly high hardware-usage bills or unrealistic requests for additional hardware. The platform should also provide visibility and controls to ensure compute resources are properly allocated and consumed by the correct users on data science teams. Visibility and controls of hardware are important, but the platform should also expose the usage of different software tools by users, for specific projects too. This level of detail helps IT leaders plan for future projects and adjust spend and tooling to be commensurate with the projects that drive the most value. It also enables IT leaders to have collaborative discussions with data science leaders on project ROI.

10. How does the platform work with traditional software development processes?

Even though data science platforms are built to enable their unique model development lifecycle, they should integrate with current software development processes. The platform should provide a workflow to enable a Dev-Test-Production schedule for the unique aspects of model development. This workflow should ensure the process captures all model assets, including code, data, comments, tools, packages, and even the development environments. Capturing all model asset information ensures that one can revert to previous model versions and promote to the latest model version in a seamless and auditable manner.