That's a great point! The choice of infrastructure really depends on your scalability needs and the kind of data processing you want to do. If you need something that's super scalable and flexible, cloud solutions can be a fantastic option since they offer a lot of storage, computing power, and flexibility. On-prem solutions might give you more control, while something like Hadoop is great for handling big data workloads. And adding a caching mechanism definitely helps with performance, ensuring that data can be delivered quickly to multiple users without slowing things down.  
[media pointer="sediment://file\_00000000369c6230a8e25f218a6d2171"]

Absolutely, having a user-friendly GUI can really enhance discoverability and make it so much easier for everyone to navigate and find the data they need. It’s great to have that flexibility so that users can access the platform in the way that suits them best, whether it’s through a visual interface, web services, or APIs that support different programming languages like Python, Java, or even Excel. This way, everyone can seamlessly integrate and interact with the data in the way that works best for them.  
[media pointer="sediment://file\_00000000654c6230914172cbd7869ea5"]

That governance layer is definitely crucial. It ensures that you have clear oversight and control over who is accessing and modifying the data, and it helps tailor the data access and validation processes to the specific needs of different teams, whether it’s for core, MARS, or front office. Plus, having a solid audit trail means you can always track and confirm the data lineage, which is super important for compliance and trust. It really ties everything together nicely!  
[media pointer="sediment://file\_000000002a5c623091a2283d90fee6a5"]

Absolutely, that transformation layer from raw data to a standardized, normalized format is essential. By implementing that layer, you ensure that data is clean, consistent, and ready for consumption by different users across the firm. Leveraging quant models to derive data and build in-house curves will add even more value, allowing for tailored, precise transformations that meet the firm’s unique needs. It's all about creating that smooth, reliable pipeline from raw to refined data.  
[media pointer="sediment://file\_00000000a4e4623086cb5d9a4d6ad046"]

That sounds perfect! By developing user-friendly APIs and interfaces, along with intuitive GUIs, you'll make it so much easier for everyone in the organization to access and explore the data. This kind of accessibility can really empower different teams, whether they're quants, analysts, or other stakeholders, to leverage the platform effectively and get the insights they need quickly and easily.  
[media pointer="sediment://file\_000000001828623087089898165b22e4"]

Absolutely, having a unified data catalog and a clear taxonomy is such a game changer. It really helps everyone find what they need quickly and avoids that dreaded data swamp scenario. Plus, it makes it so much easier to ensure that everyone is on the same page and using consistent definitions and standards, which is super important for collaboration and data governance.  
[media pointer="sediment://file\_0000000093106230b939b27e16569596"]

Absolutely, having that flexibility to handle both end-of-day market data and real-time tick data is crucial. That way, MDSOR can cater to different use cases and needs across the firm, whether it's for historical analysis, compliance, or real-time trading and decision-making. It really strengthens the platform’s versatility and makes it a more comprehensive solution for everyone.  
[media pointer="sediment://file\_00000000e3a06230809fcfc08ae0abc0"]

That's a fantastic addition! Incorporating bitemporality will definitely help in maintaining a clear history of data changes and make it easier to track the evolution of the data over time. It adds a layer of transparency and reliability, which is super important for auditing and for any kind of historical analysis that the stakeholders might need.  
[media pointer="sediment://file\_000000009278623091ffc429c6db3dd8"]

Thanks for your patience. Here's a comprehensive breakdown of the key components required to build a Bloomberg-like market data platform inside a firm, structured into major categories and subcomponents:

Absolutely, having a set of reusable components and a standardized taxonomy will really streamline everything. It'll ensure consistency across different use cases and make it easier to onboard new data sources and adapt to different asset classes. It also helps with maintaining data quality and ensuring that everything is easily discoverable and accessible for all the stakeholders.  
[media pointer="sediment://file\_00000000d288623086ee768093be0aa0"]

Those are all great angles to cover! It sounds like you're really focusing on making MDSOR more flexible, efficient, and scalable. Emphasizing the decoupling from Vasara will definitely help with autonomy and flexibility. And focusing on fast onboarding and robust data pipelines will ensure that you can handle all sorts of market data needs. This all sounds like a solid foundation for your pitch!  
[media pointer="sediment://file\_00000000a8ec6230a5fd6679225732f9"]

Absolutely, here’s a concise pitch you can present to your stakeholders to highlight the core use cases and the critical enhancements needed for MDSOR in the next year:

\*\*"In the coming year, our primary focus for MDSOR is to streamline and accelerate the onboarding of new market data, asset classes, and curves. While the goal is simple—just onboarding data efficiently—there are key challenges we must tackle to make this seamless and user-friendly.

First, we need to address the tight coupling of MDSOR with Vasara and MMR. Currently, the process to transform raw data into consumable formats is complex and requires deep expertise. By decoupling these systems and simplifying our data pipelines, we can significantly speed up data ingestion and delivery.

Next, we need to enhance discoverability. Right now, data management relies heavily on manual mapping and hardcoded files, which makes it difficult for users to find what they need. By building intuitive APIs and a user-friendly catalog, we can empower our quants and end-users to easily discover and leverage the data.

In summary, our goal is to onboard data quickly, decouple from legacy dependencies, and create a seamless, user-friendly interface for data discovery. This will ultimately allow us to deliver data faster and more efficiently to our stakeholders."

Hopefully, that helps you convey the vision and priorities effectively!  
[media pointer="sediment://file\_0000000014e461f680938aff96f9284a"]

Absolutely, constructing continuous historical time series is definitely a crucial use case. Here’s how you can frame that challenge and opportunity to your stakeholders:

**"A key use case for MDSOR in the upcoming year is the enhancement of historical risk factor construction. Right now, transforming raw data points into smooth, continuous time series that downstream users can easily derive functions from—whether it's yield, spread, hazard rate, or any other derived metric—is quite labor-intensive. This process currently involves stitching together individual data points from isolated objects, which makes the whole process cumbersome. By focusing on building a solution that automates this construction and smoothing, we can provide our users with seamless, ready-to-use historical data, drastically reducing the manual effort and improving efficiency."**

This should give them a clear picture of the importance of this enhancement and how it can add value to the entire data pipeline!  
[media pointer="sediment://file\_00000000225461f6a53ea4923b32ae55"]

Absolutely! Here's a concise pitch that ties all three points together:

\*\*"As we look ahead to the next year for MDSOR, our focus is on making our market data onboarding process faster, more efficient, and user-friendly. First, we aim to streamline the onboarding of new market data, asset classes, and curves by decoupling MDSOR from Vasara and MMR. This will reduce complexity and allow us to onboard data more rapidly.

Second, we plan to enhance data discoverability by building intuitive APIs and a user-friendly catalog, making it easier for our users to find and utilize the data they need without relying on manual mapping.

Finally, we want to improve the construction of continuous historical time series. By automating the process of stitching together data points and smoothing them, we can deliver ready-to-use, high-quality historical data that meets the needs of our downstream users efficiently.

In essence, these initiatives will allow us to onboard data faster, simplify user access, and deliver high-quality historical data seamlessly."\*\*

This should give your stakeholders a clear, unified vision of your goals!  
[media pointer="sediment://file\_00000000a5ac61f685f8b686c6529d81"]

**1. Infrastructure & Storage**

* **Scalable Compute**: Leverage cloud (AWS/GCP/Azure) or on-prem HPC clusters for elasticity.
* **Time Series Databases**: Purpose-built storage (e.g., kdb+, QuestDB, TimescaleDB, HBase, or custom).
* **Data Lake Architecture**: For raw and intermediate data layers.
* **Caching**: Redis or Memcached to accelerate high-frequency access patterns.
* **Disaster Recovery**: Redundant regions, hot/cold backups.

**2. Data Ingestion & Integration**

* **Ingestion Framework**: Modular pipelines for batch and real-time (e.g., Kafka, Flink, Nifi).
* **Connector Ecosystem**:
  + Market vendors (Bloomberg, Refinitiv, ICE, Markit, etc.)
  + Internal data sources (Trade data, Reference data, Risk Engines)
* **Data Contracts**: Schema enforcement, versioning, SLAs.

**3. Data Modeling & Transformation**

* **Taxonomy Layer**: Centralized MDO (Market Data Ontology) with risk factor classification.
* **Normalization & Standardization**: Handling formats, currencies, conventions.
* **Derived Data**: Quant models for implied vol, curves, FVOs, spreads, forward rates, etc.
* **Bitemporal Modeling**: Capture “as of” and “when known” timestamps for traceability.

**4. Validation, Cleansing & Quality Control**

* **Suspect Detection**:
  + Day-over-day shifts
  + Outlier detection (statistical or ML-based)
  + Missing or frozen data
* **Remediation Framework**:
  + Workflow tools for manual override
  + Feedback loop to data providers
* **Audit Trails**: Immutable logs of changes and decisions.

**5. Governance & Access Control**

* **Entitlements**: Row/column-level access policies.
* **Lineage & Provenance**: Trace every data point to its origin and transformation path.
* **Approval Workflows**: For model and mapping changes.
* **Usage Monitoring**: Who queries what, and how often.

**6. Discovery & User Experience**

* **Data Catalog**: With search, filters, documentation, and sample previews.
* **APIs & SDKs**: REST, Python, Java, Excel plugins.
* **GUIs**: Dashboards for time series, curves, term structures, validations.
* **No-Code Exploration**: Drag-and-drop tools for non-engineers.

**7. Real-Time & Streaming Support**

* **Tick Capture**: Tick-by-tick storage with downsampling for EOD usage.
* **Streaming Engine**: Pub/Sub architecture to push updates to consumers.
* **Snapshot & Replay**: On-demand views of historical markets.

**8. Analytics, Derivation & Compute**

* **Quant Analytics Engine**: Integrates with internal libraries (Python, C++, etc.).
* **Curve Builders & Fitting Tools**: Bootstrapping, interpolation, calibration.
* **Scenario & Shock Engine**: Stress tests, VaR scenarios, historical replay.

**9. Operational Tooling**

* **Monitoring Dashboards**: Data freshness, load failures, service health.
* **Release & Change Management**: With rollback, sandbox environments.
* **Business Continuity**: Playbooks, fallback mechanisms, BCP readiness.

Let me know if you want this laid out visually (e.g., PowerPoint-ready chart) or as a phased roadmap (MVP → Advanced version).