**Open with the Business Impact (C-Suite Hook)**

Right now, Capital relies on Mars outputs. Mars uses delta/gamma approximations fast, but not full reval.

The plan is to leverage EMR for Capital because it gives **full revaluation VaR**.

But for EMR to qualify, it must run **TWC**, and that requires 18 years of market data for every risk factor.  
The problem is: MDSOR is the engine behind EMR, and it wasn’t designed for that type of historical load at scale.

Here’s how I’d arm you for tomorrow — a **precise hybrid technical/business pitch** for the Vasara architect & head of Quant.  
This speaks his **system design + quantitative + C-suite lens** while keeping it focused, layered, and strategic.

**Pitch: Why TWC + EMR + MDSOR is a Strategic Bottleneck to Solve Now**

**1. Open with the Business Impact (C-Suite Hook)**

“Right now, Capital relies on Mars outputs. Mars uses delta/gamma approximations — fast, but not full reval.  
The plan is to leverage EMR for Capital because it gives **full revaluation VaR**. But for EMR to qualify, it must run **TWC** — and that requires 18 years of market data for every risk factor.  
The problem is: MDSOR is the engine behind EMR, and it wasn’t designed for that type of historical load at scale.”

**2. Define the Technical Constraint (Architect Lens)**

**Today:**

* Each historical date = isolated serialized Java object (no native stitching)
* 18 years of history = **4,000+ objects per risk factor**
* To run TWC → must **deserialize, stitch, and align** these objects at runtime
* Not columnar, not optimized for scan-and-summarize

**Impact:**

* High latency + memory cost for long-span analytics
* Forces manual orchestration by senior developers
* Creates dependency bottlenecks — remediation stuck waiting on MDSOR tech fixes

**3. Quantitative Angle (Risk/Model Lens)**

* **TWC** isn’t just a big data pull — it’s a **model calibration** step over 18 years
* Any inconsistency in proxies, curves, or gaps **changes the stress window** (e.g., GFC → COVID shift)
* Regulatory expectation = validate history **and** justify discrepancies
* Without optimization + SME remediation, results risk being **inconsistent or delayed**

**4. The Real Bottleneck (Strategic Lens)**

It’s **not** just:

* A staffing issue → you can’t fix this with generic developers
* A tech issue → IT fixes alone won’t meet timelines without domain SMEs

It **is**:

* A system-design gap that needs **dual-path architecture**:
  1. **Bitemporal MDSOR path** → for FO pricing & traceability
  2. **Optimized analytics path** → for historical curve scans & TWC

**5. Roadmap / Ask**

**Phase 1 — Tactical Enablement (0–3 months)**

* Place **Data/Quant SMEs** per asset class to remediate gaps while IT works
* Build tactical stitching + derived datasets for immediate TWC testing

**Phase 2 — Architectural Fix (3–12 months)**

* Implement dual-path data architecture (bitemporal + analytics store)
* Introduce **time-series optimized layer** (Parquet/TSDB) for long-span analytics

**Phase 3 — Operationalize & Scale (12+ months)**

* Integrate Capital-caliber TWC runs into EMR with automated validation
* Reduce reliance on manual intervention → free senior devs for strategic work

💡 **Soundbite for him:**

“If we design this right, EMR becomes Capital-ready without crippling FO operations. But to get there, we need **the right system pattern** and **the right SMEs** — otherwise, we’re asking MDSOR to be a Ferrari on a gravel road.”

If you want, I can turn this into a **3-slide visual** so tomorrow you can walk him from business → technical → solution in under 5 minutes, which is perfect for his style.

Do you want me to prep that now?

Here’s the **expanded and final skill set** with the **modeling, Data Strategist, and Quantitative Data Strategist** angles included — so it covers **domain, technical, architectural, and analytical modeling skills** for the kind of SMEs you need.

**Skills Required for MDSOR/TWC Advanced Data SMEs**

**1. Large-Scale Data Analytics**

* Experience handling **large historical datasets** (10–20+ years, multi-asset class)
* Ability to **extract insights from messy data** — detect systemic issues, correlations, and anomalies
* Skilled in **performance profiling** for heavy historical queries and batch analytics
* Familiarity with **parallel processing, distributed storage, and aggregation techniques**

**2. Architectural & Systems Thinking**

* Ability to **design for scale** — separating real-time pricing vs. historical analysis paths
* Can identify **system limitations** and propose architectural redesigns
* Proficient in **proof-of-concept builds** to validate solutions before full implementation
* Comfortable reading **Vasara code** and brainstorming **API Factory** concepts for standardized, reusable data access
* Familiarity with **MDSOR internals** (Java object storage, bitemporal versioning, API integrations)

**3. Tooling & Automation**

* Skilled in building **tactical tools** within MDSOR to address data gaps
* Capable of **automating reconciliation and validation** processes
* Proficient in scripting and wrangling data (Python, SQL, Java basics)
* Able to create **derived datasets** for analytics without degrading production

**4. Market Data Domain Expertise**

* **Asset class specialization**: Rates, Credit, FX, Commodities, Equities, Structured Products
* Understanding of **risk factor modeling** (curves, volatility surfaces, spreads)
* Knowledge of **proxy rules**, historical model changes, and their downstream impact
* Familiarity with **regulatory capital requirements** (GVAR, SVAR, TWC, SHRP implications)

**5. Modeling Skills**

* Understanding of **quantitative risk models** (VaR, SVaR, Expected Shortfall, stress testing)
* Ability to calibrate models using long historical datasets (e.g., 18-year TWC runs)
* Experience in **curve construction, volatility surface fitting, and scenario generation**
* Capable of validating model results and identifying data-driven model weaknesses

**6. Data Strategist Skills**

* Skilled in **data lifecycle management** — sourcing, ingestion, normalization, validation, and consumption
* Ability to design **data operating models** that ensure quality and scalability
* Strong governance knowledge (data lineage, traceability, audit-readiness)
* Can **translate business objectives into actionable data projects**

**7. Quantitative Data Strategist Skills**

* Ability to **bridge between quantitative modeling and data engineering**
* Experience aligning **data architecture** with **model requirements**
* Skilled at **identifying key risk drivers** and ensuring data completeness for them
* Able to guide **model-ready data transformations** to minimize manual intervention
* Strong ability to work with quants to **prototype and backtest model improvements**

**Final Soundbite for your manager:**

“These aren’t just data people — they’re a hybrid of market data SMEs, quantitative model thinkers, and system architects. They can navigate MDSOR’s technical limitations, build tactical tools, guide IT’s build-out, and ensure our models and Capital processes have the right data, at the right quality, at the right time.”

If you want, I can now break this into **three role archetypes** (Pure Data SME, Hybrid Data/Tech SME, Quantitative Data Strategist) so hiring discussions are laser-focused and aligned to priorities.  
Do you want me to prepare that?

Here’s a concise **Executive Summary** you can use for your TWC EMR discussion:

**Executive Summary – TWC EMR Situation**

The **MDSOR for EMR solution** began with a **narrow focus** — IR and FX risk factors for internal management purposes.  
However, expanding its use to support **TWC** has exposed **critical system limitations** in areas such as:

* **Validation** – Inadequate tooling for large-scale, date-by-date checks
* **Stitching** – No native logic to link daily objects into coherent long-range series
* **Proxy Management** – Configurations are hard-coded, inflexible, and hard to trace
* **Data Quality Insight** – Data is present but often incorrect; locating and fixing issues is slow and manual

**Operational Reality**

* **Fixing issues requires an army** – The scale (millions of objects) and complexity demand skilled, hands-on work.
* **Clean PnL priorities** mean **Tech resources are fully loaded**; they can’t pivot to address systemic gaps without impacting regulatory deliverables.
* **IT dependency is risky** – Talent quality is inconsistent; many hires lack the skill depth needed for bitemporal, object-based systems.
* **Data Fabric is unsuitable for time series**; Carina UI is unusable for multi-asset, cross-date analysis.

**Impact on Current State**

* Our **best SMEs** are pulled into remediation work instead of focusing on strategic initiatives.
* Deliverables cannot be paused, so fixes happen in parallel with ongoing production demands — an unsustainable approach.

**What’s Needed**

1. **Specialized Staffing** – Asset-class SMEs who understand both system architecture and risk factor modeling.
2. **Time Allocation** – Space to fix systemic gaps without compromising deliverables.
3. **Skill Depth** – People who can navigate Vasara code, debug Java objects, and design tactical solutions.
4. **Reduced Reliance on IT** – Build in-house functional/technical SME capacity to own outcomes.

If you want, I can turn this into a **one-slide “C-suite clarity” graphic** showing **scope creep → limitations → operational strain → staffing need** so your manager can internalize it quickly.  
Do you want me to do that?

* The MDSOR for EMR solution began with a narrow focus — IR and FX risk factors for internal management purposes. Expanding to support TWC exposed critical limitations:
* • Validation – Inadequate tooling for large-scale, date-by-date checks
* • Stitching – No native logic to link daily objects into coherent series
* • Proxy Management – Hard-coded configurations, inflexible, hard to trace
* • Data Quality Insight – Data present but often incorrect; finding/fixing is slow
* Operational Reality:
* • Fixing issues requires skilled, hands-on work across millions of objects
* • Clean PnL priorities overload Tech; systemic fixes delayed
* • IT dependency is risky; skill depth inconsistent
* • Data Fabric unsuitable for time series; Carina UI poor for cross-date analysis
* Impact:
* • SMEs pulled from strategic work to remediation
* • Fixes happen alongside production deliverables – unsustainable
* What’s Needed:
* 1. Specialized staffing (asset-class SMEs with architecture + modeling skills)
* 2. Time allocation for systemic fixes
* 3. Skill depth in Vasara, Java, debugging, tactical solutions
* 4. Reduce reliance on IT; build in-house functional/technical SME capacity