

A hybrid, unified platform and comprehensive set of tools

AI in a Hybrid World

Challenges and Architectural Approaches in Financial Services

Aric Rosenbaum

Red Hat

Chief Technologist, Global FSI

aric@redhat.com



Aric Rosenbaum

Red Hat, Chief Technologist, Global FSI



e: aric@redhat.com

m: +1-973-610-4671

Background

- Aric serves as the Chief Technologist in Red Hat's Global FSI team, where he helps clients meet their strategic priorities through the use of open source technology.
- Prior to joining Red Hat, he led large, digital transformation projects at Goldman Sachs' Investment Management Division and was co-founder/CTO of several FinTechs in equity and FX trading.
- In his 20+ years of experience, Aric has designed and built multiple front, middle and back office systems and deployed his first app to the cloud in 2008.

Area of Expertise

Financial services, capital markets, FinTech, microservices, continuous integration / delivery, agile, cloud computing, hybrid cloud

Selected Experience

At Goldman Sachs, Aric co-led a global team of 60+ engineers building the next generation client lifecycle management system.

Aric is a member of the Governing Board of the Fintech Open Source Foundation (FINOS).



Tech will Transform

Emerging tech, like AI, will transform financial services by improving customer experience, reducing risk and making firms more efficient

Tailored Advice Incorporates Bank Insights and Offerings

Gen AI wealth management models improve UX while delivering suitable and compliant portfolios

AR

As a married 40 year old stay at home dad with two children (ages 12 and 18) and a wife who makes \$200,000 a year, design an investment portfolio.

AR

As a widowed 80 year old with grown children and no income, design an investment portfolio.

Example allocation:

- U.S. Total Stock Market Index Fund: 40%
- International Stock Market Index Fund: 20%
- Bond Market Index Fund: 30%
- Real Estate Investment Trust (REIT) Fund: 10%

With links to products,
performance and
disclosures

4. College Savings (529 Plan):

Since you have two children, consider opening a 529 college savings plan for each child. These plans offer tax benefits for education expenses. Invest in age-based portfolios or select a diversified mix of stock and bond funds based on your risk tolerance and the time horizon until your children start college.

5. Risk Tolerance:

Your risk tolerance is a critical factor in portfolio design. Given your age and the fact that your wife has a stable income, you may be comfortable with a moderate to moderately aggressive portfolio. However, it's essential to consider how comfortable you are with market fluctuations and your long-term financial goals when determining your risk tolerance.

2. Fixed-Income Investments:

The core of your portfolio should consist of fixed-income investments, which provide regular income and stability. Consider the following options:

- Treasury Bonds: Invest in U.S. Treasury bonds, which are considered one of the safest investments.
- Municipal Bonds: Municipal bonds can offer tax advantages and steady income.
- Corporate Bonds: Choose highly rated corporate bonds for higher yields but with some risk.
- Certificates of Deposit (CDs): Consider laddering CDs to have access to funds at different intervals.

Allocate a significant portion of your portfolio, perhaps 60-70%, to these fixed-income investments to generate income and preserve capital.

3. Dividend-Paying Stocks:

To supplement your income and potentially benefit from capital appreciation, allocate a smaller portion (around 20-30%) of your portfolio to dividend-paying stocks from established, dividend-paying companies. Look for companies with a history of consistent dividend payments.

Accelerating AppDev by empowering SMEs

Reduced friction allows users to more easily, accurately and efficiently create requirements

AR

In the format of given, when, then, write a test script for an application that checks if a customer is over 18 years old.



Certainly! Here's a test script for an application that checks if a customer is over 18 years old, following the "Given, When, Then" format commonly used in behavior-driven development (BDD) and test automation:

Test Script: Check Customer Age Verification

Given the customer is on the registration page

- * Navigate to the registration page of the application

When the customer enters their birthdate

- * Input the customer's birthdate in the appropriate field

And the customer submits the registration form

- * Click the "Submit" button on the registration form

Then the application should validate the customer's age

- * If the customer's age is less than 18, display an error message
- * If the customer's age is 18 or older, proceed with the registration

And the customer should be registered successfully

- * Verify that the customer is registered successfully if age is 18 or older
- * Ensure the customer is not registered if age is less than 18

And the customer should be redirected to the welcome page

- * If registered successfully, verify that the customer is redirected to the welcome page

And the customer should receive a confirmation email

- * Check if a confirmation email is sent to the customer's provided email address

And the customer's age should be stored in the database

- * Verify that the customer's age is correctly stored in the database

And the customer's information should be accurate

- * Ensure that other customer information (e.g., name, email) is correctly captured and stored

And the customer should not be registered if the age is invalid

- * Confirm that the registration fails and an appropriate error message is displayed if the age is invalid

Accelerating AppDev by empowering SMEs and Developers

And then turn those requirements into code

Certainly! Here's a test script for an application that checks if a customer is over 18 years old, following the "Given, When, Then" format commonly used in behavior-driven development (BDD) and test automation:

Test Script: Check Customer Age Verification

Given the customer is on the registration page

- * Navigate to the registration page of the application

When the customer enters their birthdate

- * Input the customer's birthdate in the appropriate field

And the customer submits the registration form

- * Click the "Submit" button on the registration form

Then the application should validate the customer's age

- * If the customer's age is less than 18, display an error message
- * If the customer's age is 18 or older, proceed with the registration

And the customer should be registered successfully

- * Verify that the customer is registered successfully if age is 18 or older
- * Ensure the customer is not registered if age is less than 18

And the customer should be redirected to the welcome page

- * If registered successfully, verify that the customer is redirected to the welcome page

And the customer should receive a confirmation email

- * Check if a confirmation email is sent to the customer's provided email address

And the customer's age should be stored in the database

- * Verify that the customer's age is correctly stored in the database

And the customer's information should be accurate

- * Ensure that other customer information (e.g., name, email) is correctly captured and stored

And the customer should not be registered if the age is invalid

- * Confirm that the registration fails and an appropriate error message is displayed if the age is invalid



```
java Copy code

import org.junit.Assert;
import org.junit.Test;

public class CustomerRegistrationTest {

    @Test
    public void testCustomerAgeVerification() {
        // Given the customer is on the registration page
        navigateToRegistrationPage();

        // When the customer enters their birthdate
        String birthdate = "2000-01-01"; // Example birthdate
        enterBirthdate(birthdate);

        // And the customer submits the registration form
        submitRegistrationForm();

        // Then the application should validate the customer's age
        boolean isAgeValid = isCustomerAgeValid(birthdate);

        // And the customer should be registered successfully
        if (isAgeValid) {
            Assert.assertTrue(isCustomerRegisteredSuccessfully());
        } else {
            Assert.assertFalse(isCustomerRegisteredSuccessfully());
        }

        // And the customer should be redirected to the welcome page
        if (isAgeValid) {
            Assert.assertTrue(isRedirectedToWelcomePage());
        }
    }
}
```

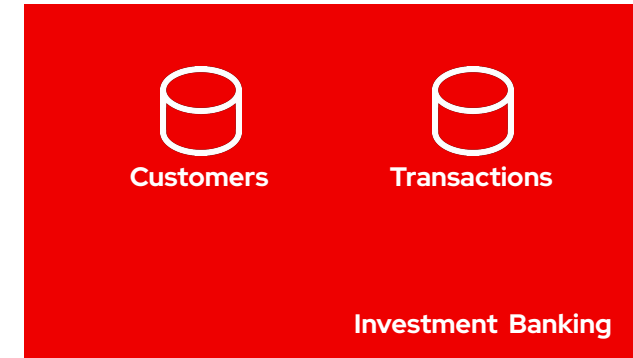
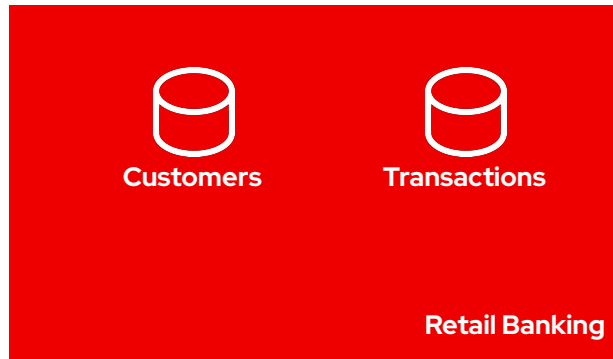

But Challenges Exist

In a highly regulated, global business, internal and external challenges exist

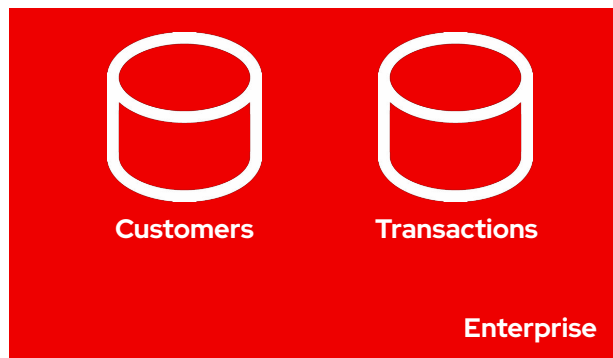
Data is Siloed

Resulting in an incomplete view, lower quality models and / or reduced efficiency

Data might be siloed by business unit:



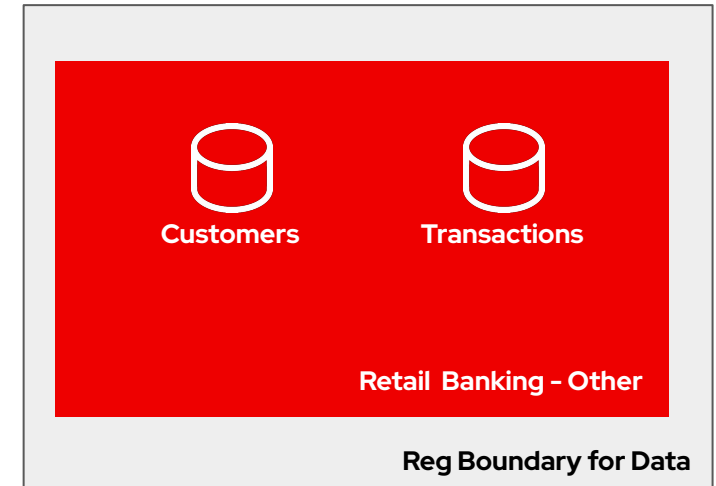
When we want this:



Data is Siloed

Resulting in an incomplete view, lower quality models and / or reduced efficiency

Or, geo:

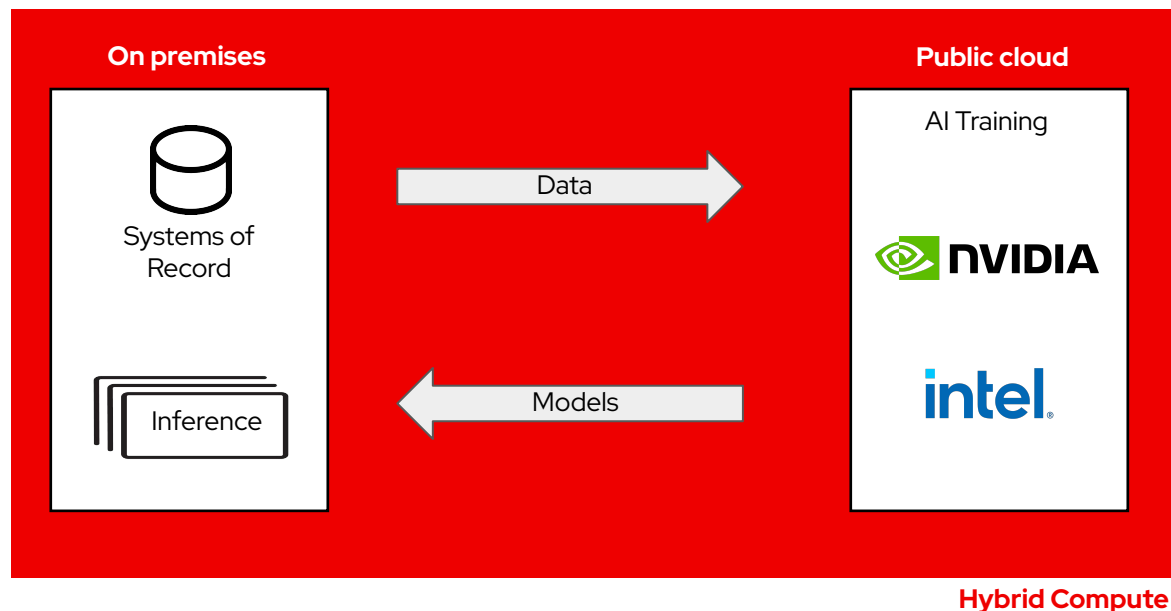


When we want this:



Requirements Leverage Resources Across Datacenters

Do we train in the cloud?

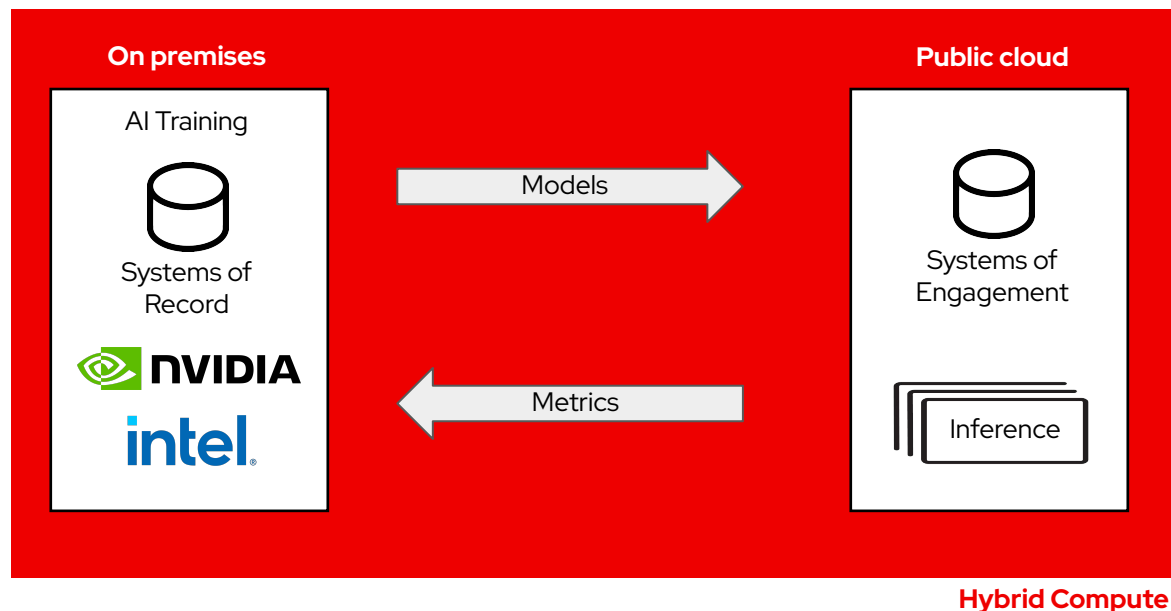


Considerations

- Systems of record (core banking, payments, trading, etc.) still sit on-prem
- Latest generation of GPUs are supply chain constrained
- Desire to always have access to leading GPUs
- Security concerns about high PII data in cloud
- Data sovereignty
- Operational efficiency
- Development velocity

Requirements Leverage Resources Across Datacenters

Or do we train on-prem?



Considerations

- Systems of record (core banking, payments, trading, etc.) still sit on-prem
- Latest generation of GPUs are supply chain constrained
- Desire to always have access to leading GPUs
- Security concerns about high PII data in cloud
- Data sovereignty
- Operational efficiency
- Development velocity

Regulators Are Focused on Concentration Risk

Similar initiatives in Canada, Europe and UK



[Home](#) > [Press](#) > [Press releases](#)

Council of the EU Press release 28 November 2022 14:30

Digital finance: Council adopts Digital Operational Resilience Act

Given the ever-increasing risks of cyber attacks, the EU is strengthening the IT security of financial entities such as banks, insurance companies and investment firms. Today the Council adopted the Digital Operational Resilience Act (DORA) which will make sure the financial sector in Europe is able to **stay resilient through a severe operational disruption**.



We live in uncertain times. Banks and other companies which provide financial services in Europe already have plans in place for their IT security, but we need to go one step further. Thanks to the harmonised legal requirements which we adopted today, our financial sector will be better able to continue to function at all times. If a large-scale attack on the European financial sector is launched, we will be prepared for it.

— Zbyněk Stanjura, Minister of Finance of Czechia

“ DORA sets uniform requirements for the security of network and information systems of companies and organisations operating in the financial sector as well as critical third parties which provide ICT (Information Communication Technologies)-related services to them, such as cloud platforms or data analytics services. DORA creates a regulatory framework on digital operational resilience whereby **all firms need to make sure they can withstand, respond to and recover from all types of ICT-related disruptions and threats.** ”

Regulators Are Focused on Concentration Risk

And US Treasury has formed a Public/Private Steering Group (CESG)



The Financial Services Sector's Adoption of Cloud Services

U.S. Department of the Treasury



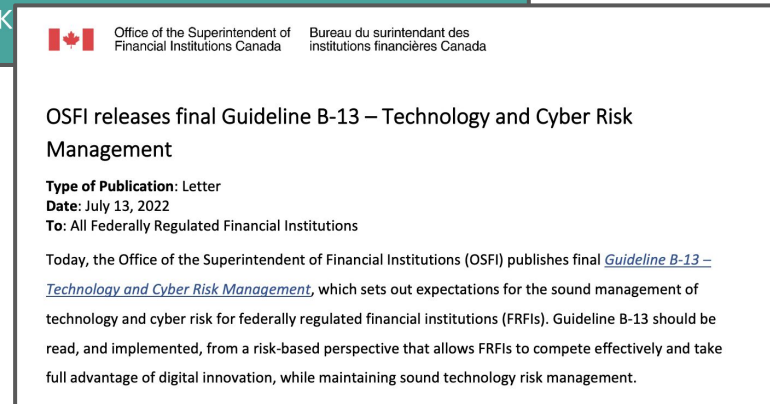
6.4 POTENTIAL IMPACT OF MARKET CONCENTRATION IN CLOUD SERVICE OFFERINGS ON THE SECTOR'S RESILIENCE

As discussed elsewhere in this report, there is evidence that the financial sector's adoption of cloud services is notable and growing, particularly with the three major CSPs: AWS, GCP, and Microsoft Azure. A large system failure or data breach at one of these CSPs could impact multiple financial institutions or U.S. consumers, though there are open questions about the extent of that impact. Such an incident could take several forms, including:

- A service interruption or degradation in performance to a single systemic financial institution or financial market infrastructure that depends on cloud services for functions critical to the financial sector;
- A service interruption or degradation in performance to a significant segment of smaller financial institutions that depend on cloud services for material business lines; or
- An interruption or degradation to cloud services that a significant number of financial institutions rely on for critical functions or material business lines. Additionally, a widespread incident could affect other service providers used by financial institutions that also rely on cloud services.

And Cyber Risk

From the US to Canada and Europe



And on the Risks of AI

With the EC proposing a set of harmonized rules



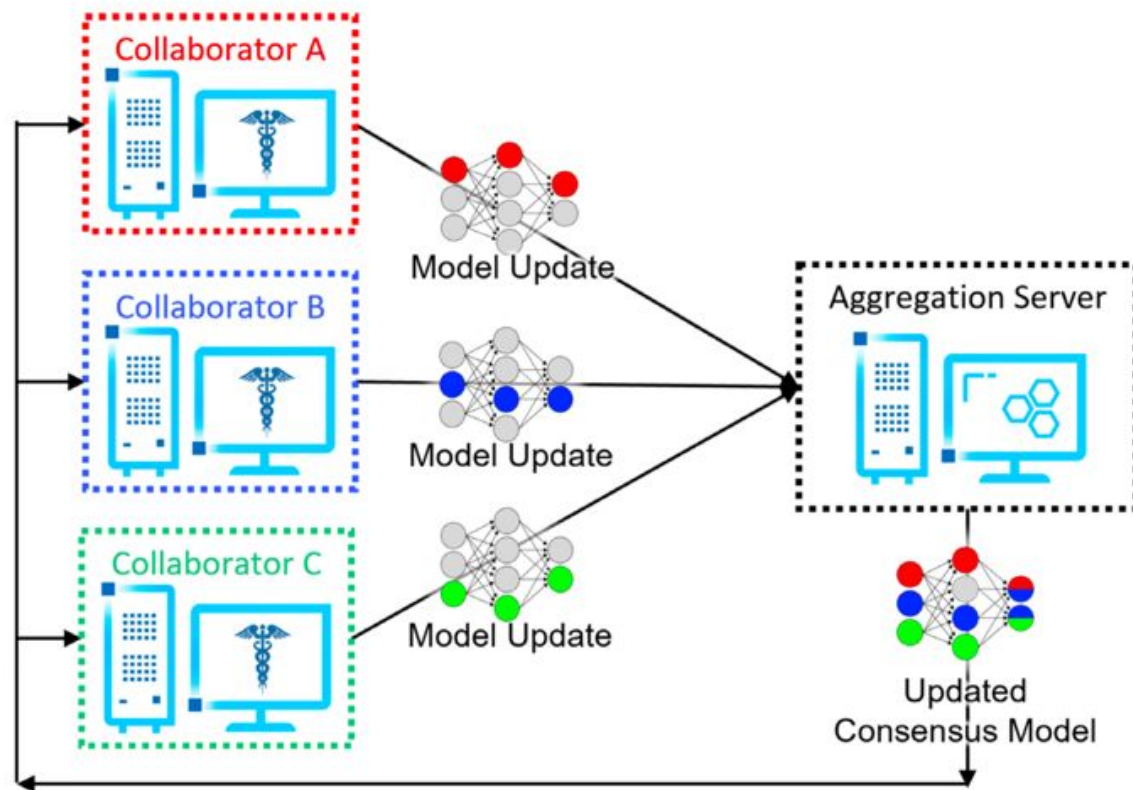
“The rapid advancement, proliferation, and transformative nature of the artificial intelligence (AI) technology has accentuated the need to consider **ethical, legal, financial and social implications** of its development and deployment. This is where **well-designed risk management practices** come in, comprised of robust testing and evaluation frameworks, implementation of clear and transparent decision-making processes, and **creation of mechanisms for accountability and redress in the event of harm.**”

Tech to the Rescue

Thankfully, we can use technology and appropriate architectural patterns to meet our requirements

Pattern #1: Train AI Models Locally without Moving Data

With Federated Learning, data is trained locally and models are aggregated



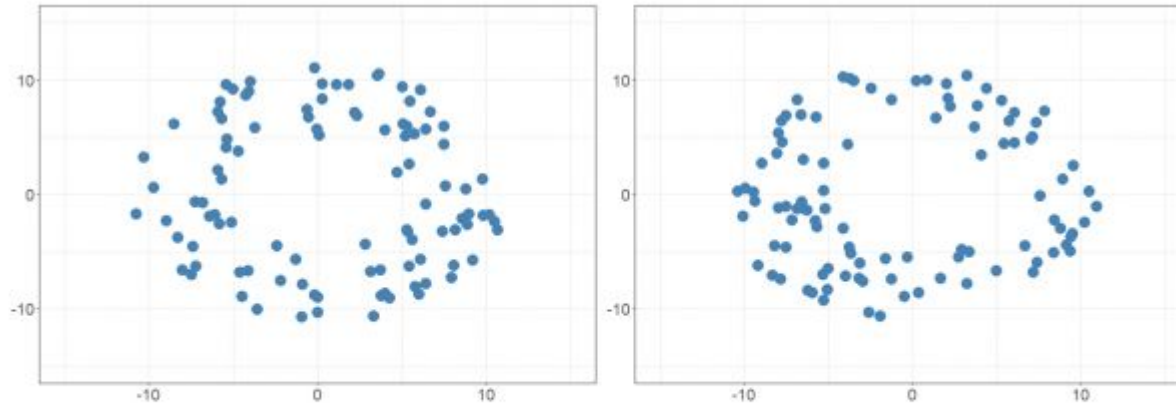
Executive Summary

- Data never leaves source site
- Only models are shared with Aggregator
- Aggregator then combines each model
- Repeat as needed
- In use by SWIFT to train their fraud anomaly detection engine



Pattern #2: Privacy Preserving Synthetic Data

Share data across regulatory boundaries and outside your enterprise



Original data

Synthetic data

The synthetic data retains the structure of the original data but is not the same

Executive Summary

- Generated synthetic data is designed to match the structure of the source data
- Can be generated locally for use by global training
- Limits risk of sharing PII internally
- Biases can be reduced
- Results must be evaluated

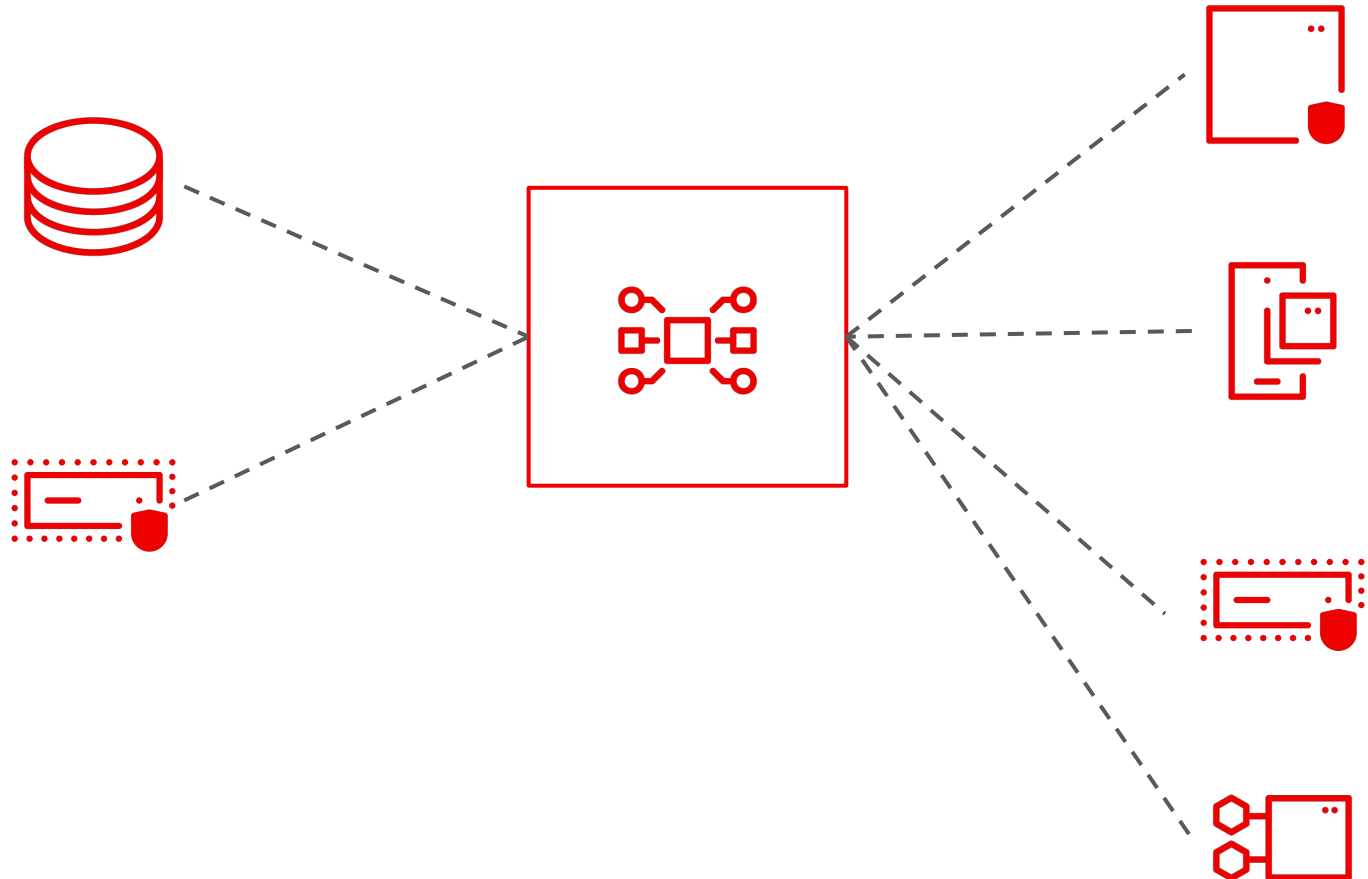
MOSTLY AI

Pattern #3: Real-Time API Proxy with Entitlements

Proxy enforces data visibility through entitlements

Producers

Consumers



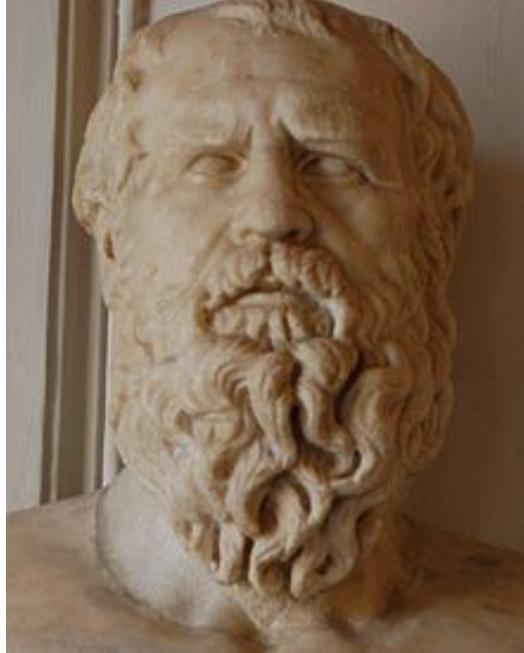
Executive Summary

- No CXL or data latency since data is not copied
- Helps to reduce risk associated with data sovereignty
- Returned attributes can be encrypted, tokenized or masked
- Dynamic in nature



Pattern #4: Preserve Optionality

Technology capabilities and regulatory oversight is changing rapidly



“ Everything changes and
nothing stands still. ”

- Heraclitus of Ephesus
(535 BC - 475 BC)

Executive Summary

- Encapsulate, encapsulate, encapsulate
- Workloads may move between on-prem and cloud as well as among hyperscalers
- Be thoughtful about proprietary APIs and services
- Open source can guard against vendor lock-in
- Copyright infringement is still an open issue (i.e. NYT)
- Expect new guidance / regulation around concentration risk, cyber security and responsible AI

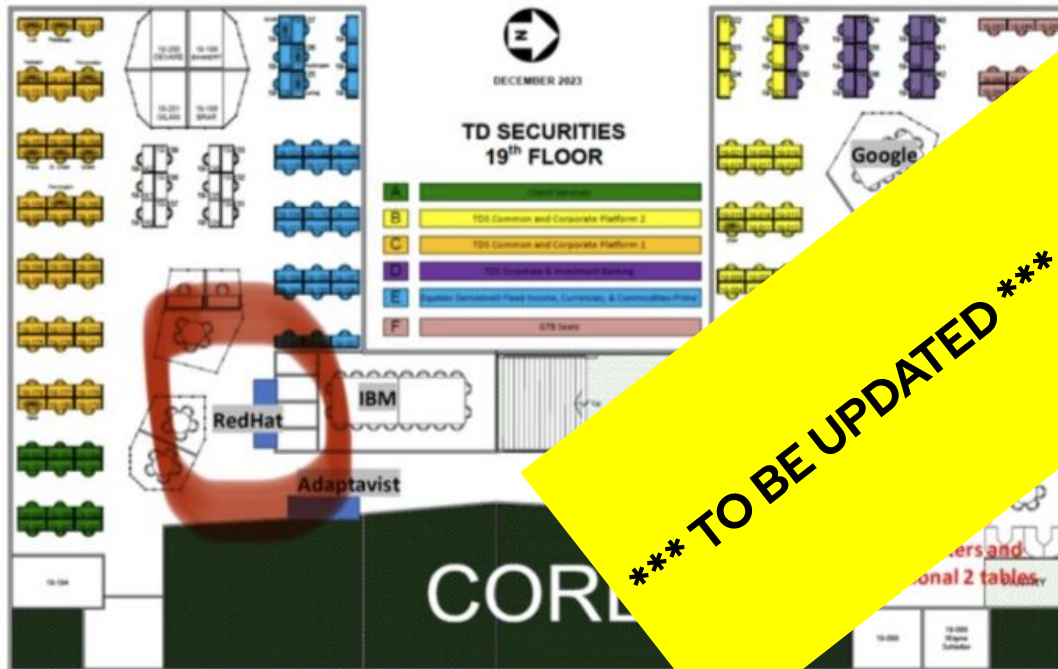
Thank You

Please visit us on the 19th Floor of 125 Park to continue the conversation



e: aric@redhat.com

m: +1-973-610-4671



Red Hat
Enterprise Linux

Red Hat
OpenShift


 **Red Hat**
Ansible Automation
Platform

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

 [linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)

 [facebook.com/redhatinc](https://www.facebook.com/redhatinc)

 [youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)

 twitter.com/RedHat