WSO2CON 2025

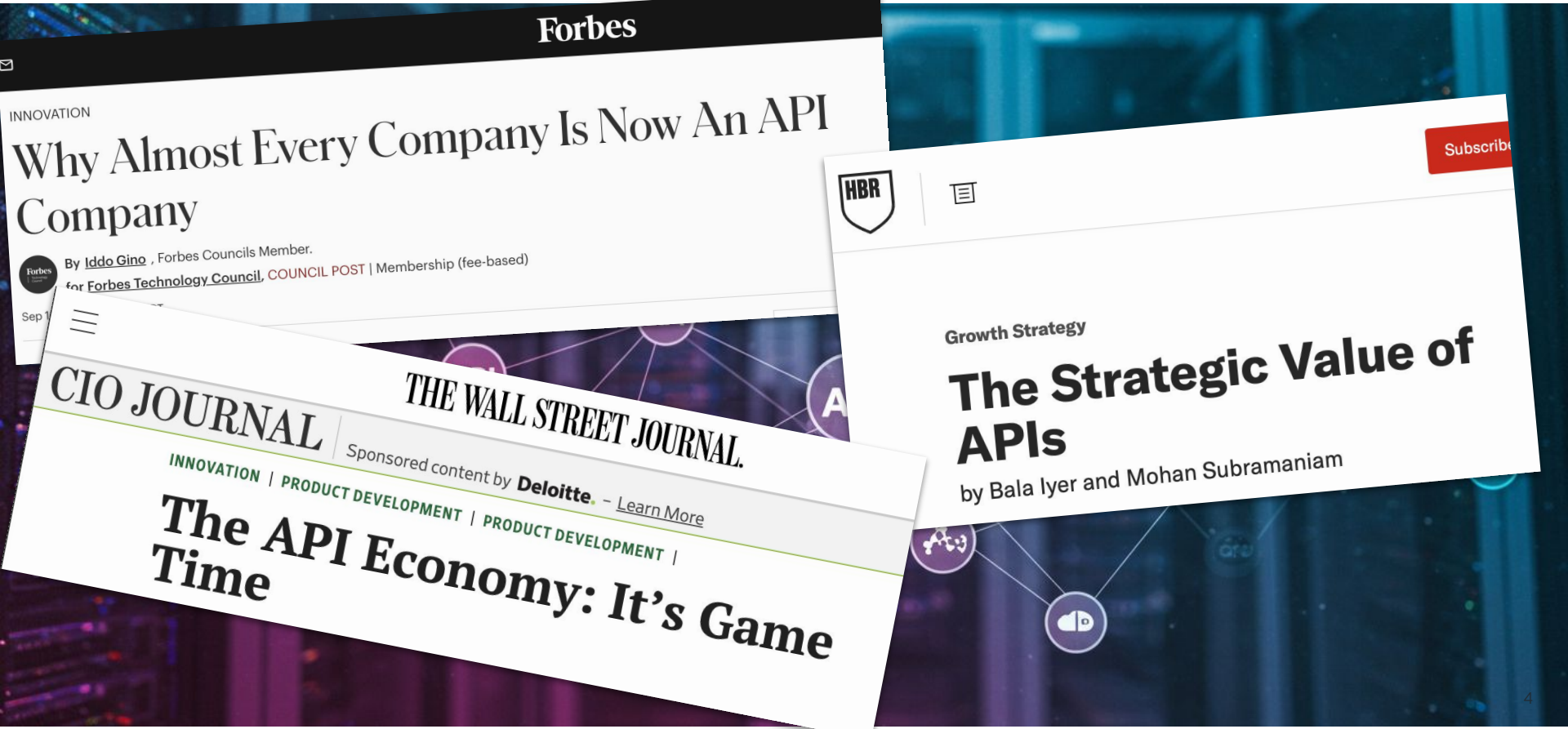
Advancing API Governance for Scalable Systems

Vidura Gamini Abhaya, Ph.D.
VP Solution Architecture & Field CTO
WSO2

Modern Organizations with Digital Channels

API Usage in Modern Organizations

API Usage in Modern Organizations



API Usage in Modern Organizations

Average Number of API endpoints per Organization – 613

(State of API Security 2024 - Imperva)

55% of the Organizations have at least 500 APIs

(2025 Global State of API Security - Traceable)

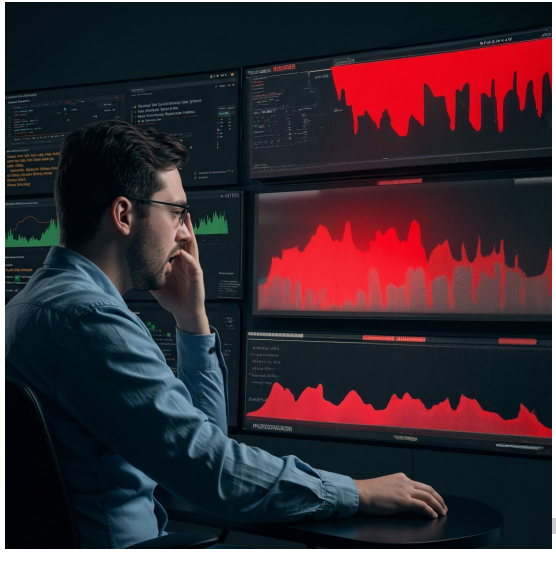
Any organization that uses digital channels are using scalable systems

Characteristics of Scalable Systems

- Powered by a large number of APIs
- Large User base
- Performance levels
- Zero Downtime

Challenges posed by Scalable Systems

- Powered by a large number of APIs
  - API Sprawl - Inconsistent API Styles,design and implementation standards, duplicate functionality and increased security vulnerabilities
  - Broader attack surface
  - Management overhead, ensuring seamless inter-service communication
  - Tracking and Managing API usage and dependencies
- Large user base
  - Multiple entry (attack) vectors
  - Authentication and Authorization challenges
- Performance levels
  - Implementations by different teams may lead to bottlenecks
- Zero Downtime
  - Changes to APIs may disrupt system uptime



Challenges posed by Scalable Systems

Each API that powers these systems need to be designed and implemented with scalability, performance and security in mind

This can only be ensured through API Governance !

API Governance (and it's undeniable need)

API Governance is the systematic management of APIs through their entire lifecycle. Governance practices includes processes, policies, standards and tools that organizations use to design,develop, deploy and manage APIs

In modern API development, governance is not about controlling and restricting the API design or usage, but *enabling teams to be innovative*, *encourage collaboration* and be agile while ensuring security, compliance, consistency are achieved, all of which is important part of an organization's strategic objectives

Modern API Management = API Governance

API Governance in Scalable Systems

Advancing API Governance for Scalable Systems

- Establish clear guidelines, standards and policies for creating APIs
  - API design principles, security protocols, business models and Terms of Service
- API First development and well defined API lifecycle
- Maintain a clear inventory of APIs and services
- Promote discovery and reuse of existing APIs
- Track API dependencies and API Usage
- Clearly defined roles and responsibilities
- Trust but verify

Advancing API Governance for Scalable Systems

Design-time Governance

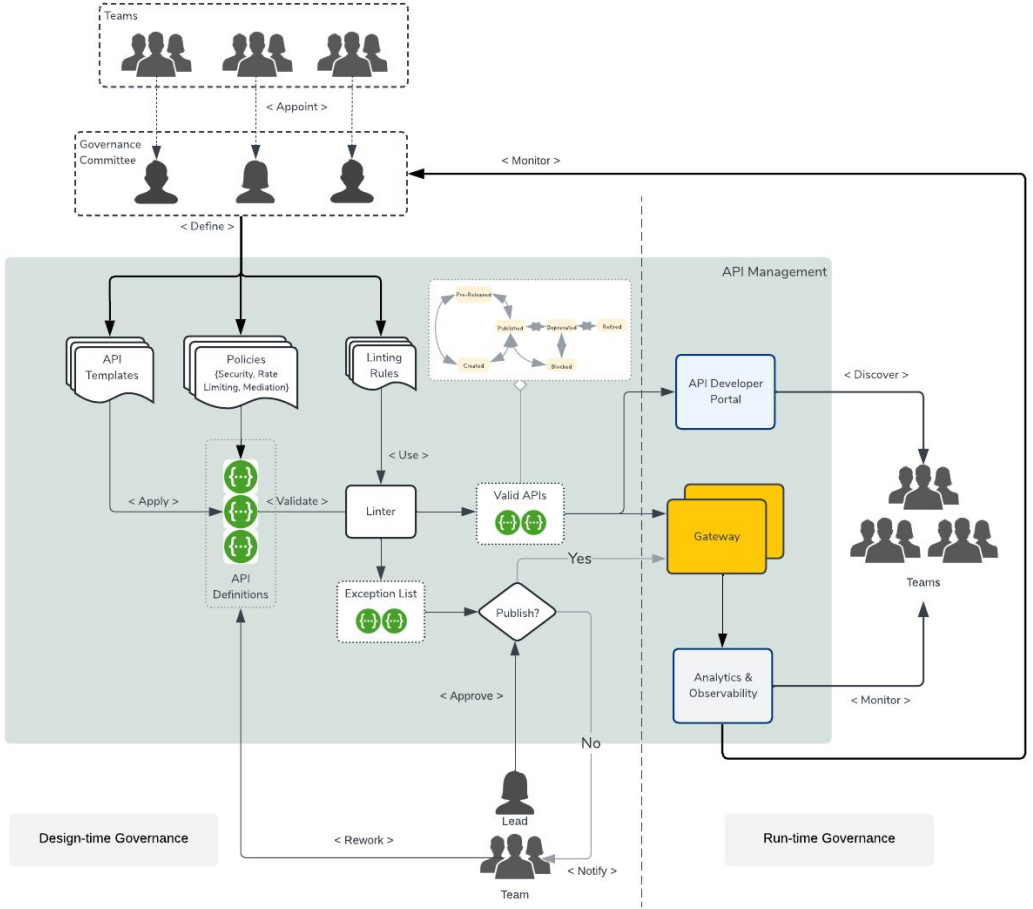
- Standards and Guidelines
- Security and Access Control
- Compliance and Legal Standards
- API Design Tools and Automation
- Reviews and Workflows

Run-time Governance

- Security and Access Control
- Traffic Management
- Monitoring and Observability
- Versioning and Deprecation

Best Practices

- Establish a Governance Committee
- Leverage Automation
- Introduce Governance Iteratively
- Continuous Monitoring and Feedback
- Invest in Education and Training



Governing AI APIs

As organizations increasingly expose AI models through APIs, governance is essential to:

- Maintain the quality and reliability of AI-powered services
- Ethical and responsible usage of AI
- Protect sensitive data used by AI models
- Control access to AI resources

Use of AI in Advancing API Governance

- Automated API Design, Creation and Deployment
- API Documentation and SDK Generation
- Identify and Manage Common Business Objects
- Intelligent Versioning and Retirement of APIs
- Predictive Threat Modeling
- Automated Security Tests
- Monitor sensitive data on network
- Performance Monitoring and Optimization
- Predictive Maintenance and Automatic Incident Response

Real World Examples – Successful API Gov implementations

- Early period of Uber
  - Monolithic architecture with different teams creating APIs
  - No governance practices

Uber

This resulted in

- Inconsistent API design across services
- Duplicated functionality and redundant APIs
- Poor Documentation and discoverability
- Security vulnerabilities due to inconsistent authentication
- Performance bottlenecks as request volumes grew

Moved to a Microservices architecture with API Gateway and Governance tools

Transition took years, cost millions

Real World Examples – Successful API Gov implementations

- Initially APIs were built in an ad-hoc manner

This resulted in

- Fragmented APIs that were difficult to maintain across different device platforms
- Inconsistent performance across their expanding global footprint
- Difficulty scaling to handle exponentially growing traffic
- Challenges integrating third-party content providers
- Reliability issues that caused service disruptions during peak viewing times

Netflix underwent a massive re-architecture effort moving to microservices and standardizing API governance. Implemented standards across their ecosystem

WSO2 Customers

Key Takeaways

Key Takeaways

- Modern organizations use scalable systems powered by thousands of APIs
- The proliferation of APIs could result in significant security vulnerabilities, performance issues and service interruption, if not managed properly
- API Governance can play a significant role in bringing order to this chaos
- Standards and practices can be introduced iteratively with the buy-in from all stakeholders
- A governance framework can be established overseen by a committee that represents all stakeholders
- AI and Automation can assist in increasing operational efficiency

Interested in knowing more ?

- Further Reading - Optimizing API Governance in a Decentralized Architectural Landscape


Thank you!
WSO2CON2025