Hexa + IDQL

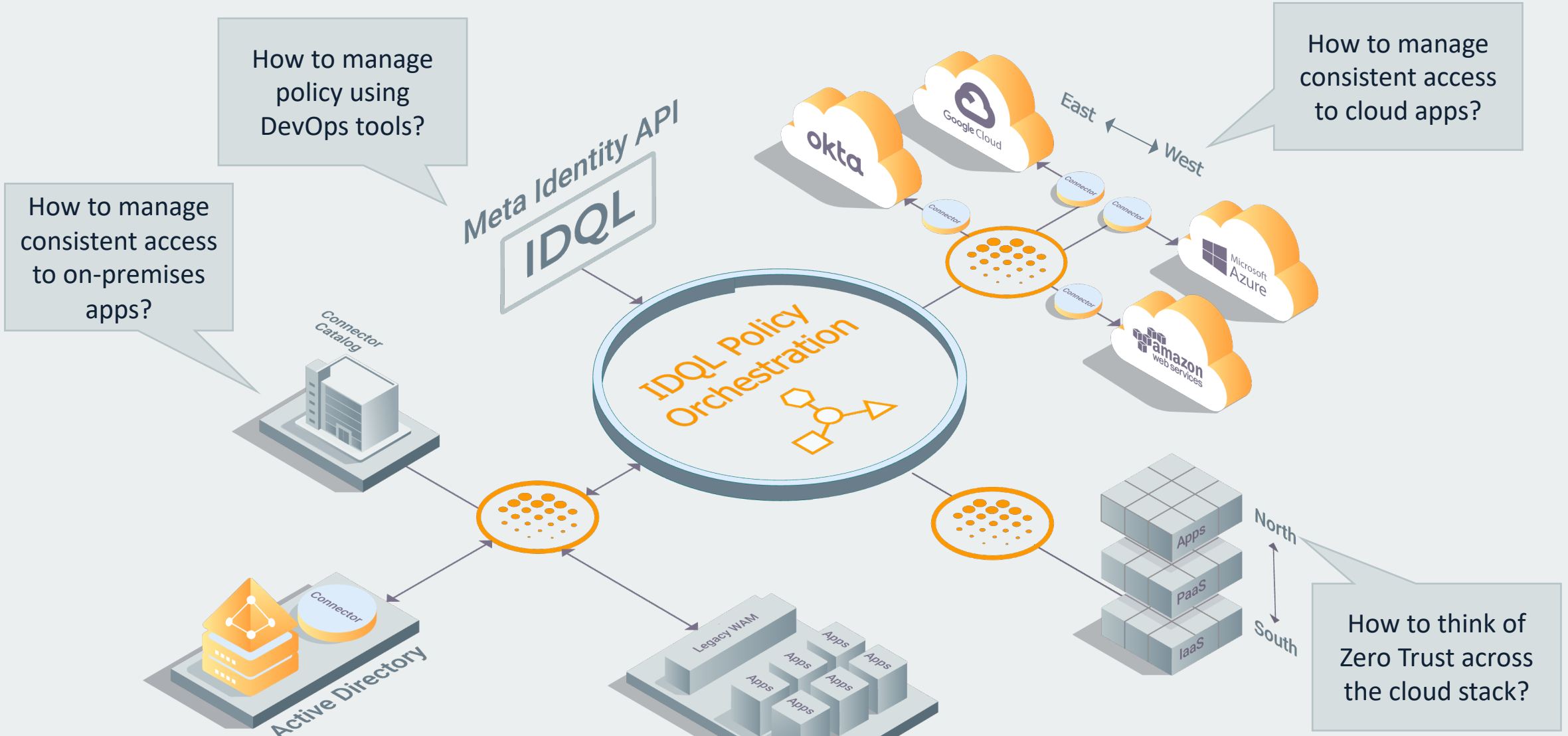
Hexa Policy Orchestration and the Identity Query Language (IDQL)

February 2022

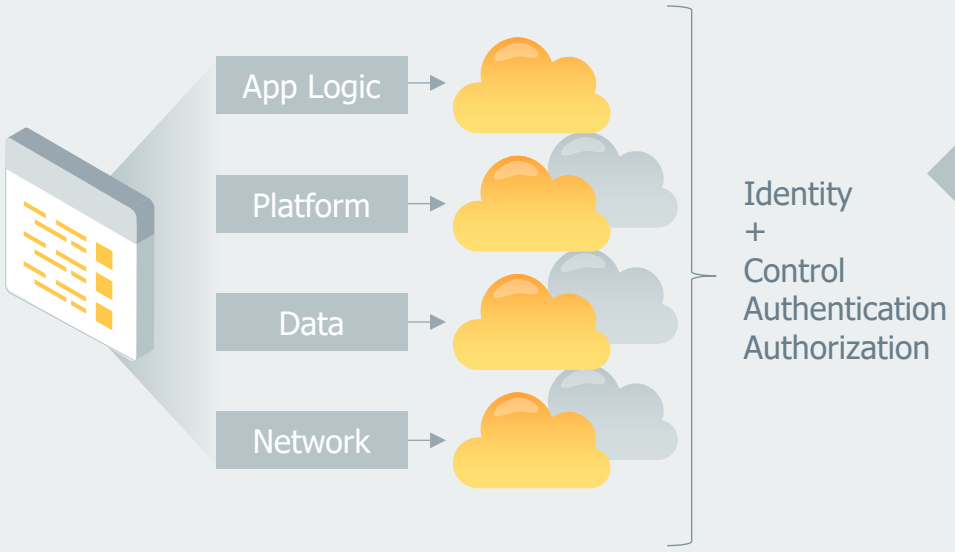
Hexa and IdentityQL – IDQL

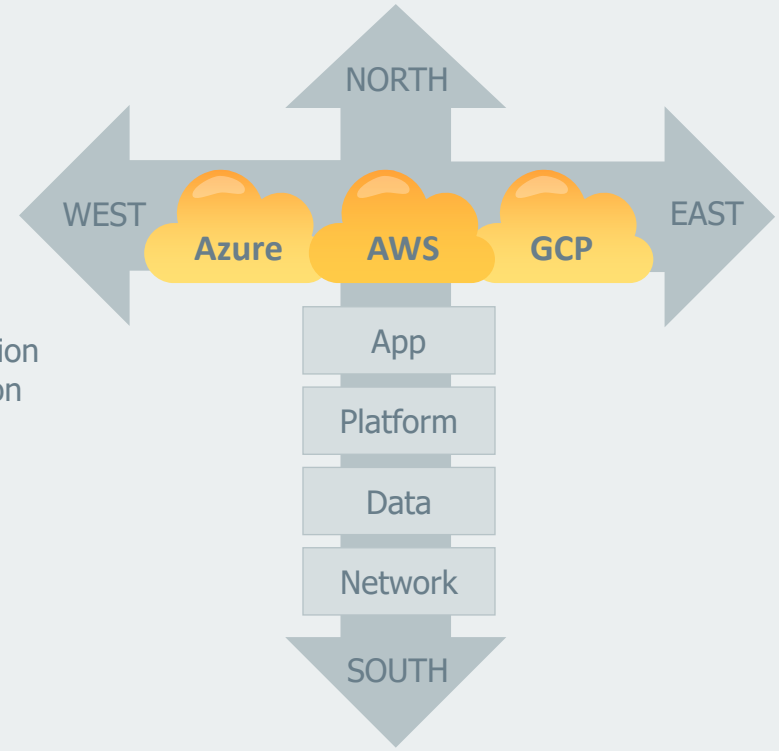
Standardizing Access Policy Across The Cloud and Across The Stack

Identity and Access Policy is Fragmented



Trends





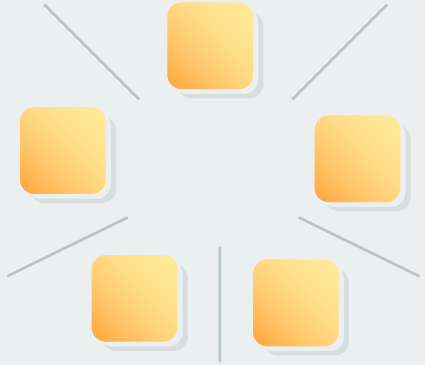
App multi cloud stack

Distributed across clouds

Zero Trust Architecture

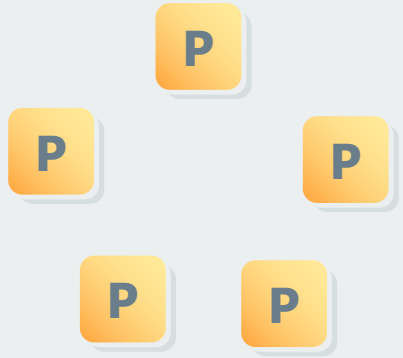
1. Distributed systems are everywhere
2. Multi-cloud apps require distributed identity
3. Multi-cloud scales along both an "East/West" and North/South" through stack

Challenges



Fragmentation across proprietary distributed systems

- Security exposure
- Need to manually manage → human error
- Poor user experience



Policies are critical but obscured & hidden

- Proprietary syntax
- Imperative hides it
- Hard to audit or understand

SCIM
XACML
SAML

Early efforts were built for another purpose

- Required too much work
- Platforms didn't adopt
- Apps didn't adopt

IDQL and Hexa – Open Source Policy Orchestration

Hexa

IDQL
Universal Declarative Policy

IDQL Policy Details v0.1

Subject

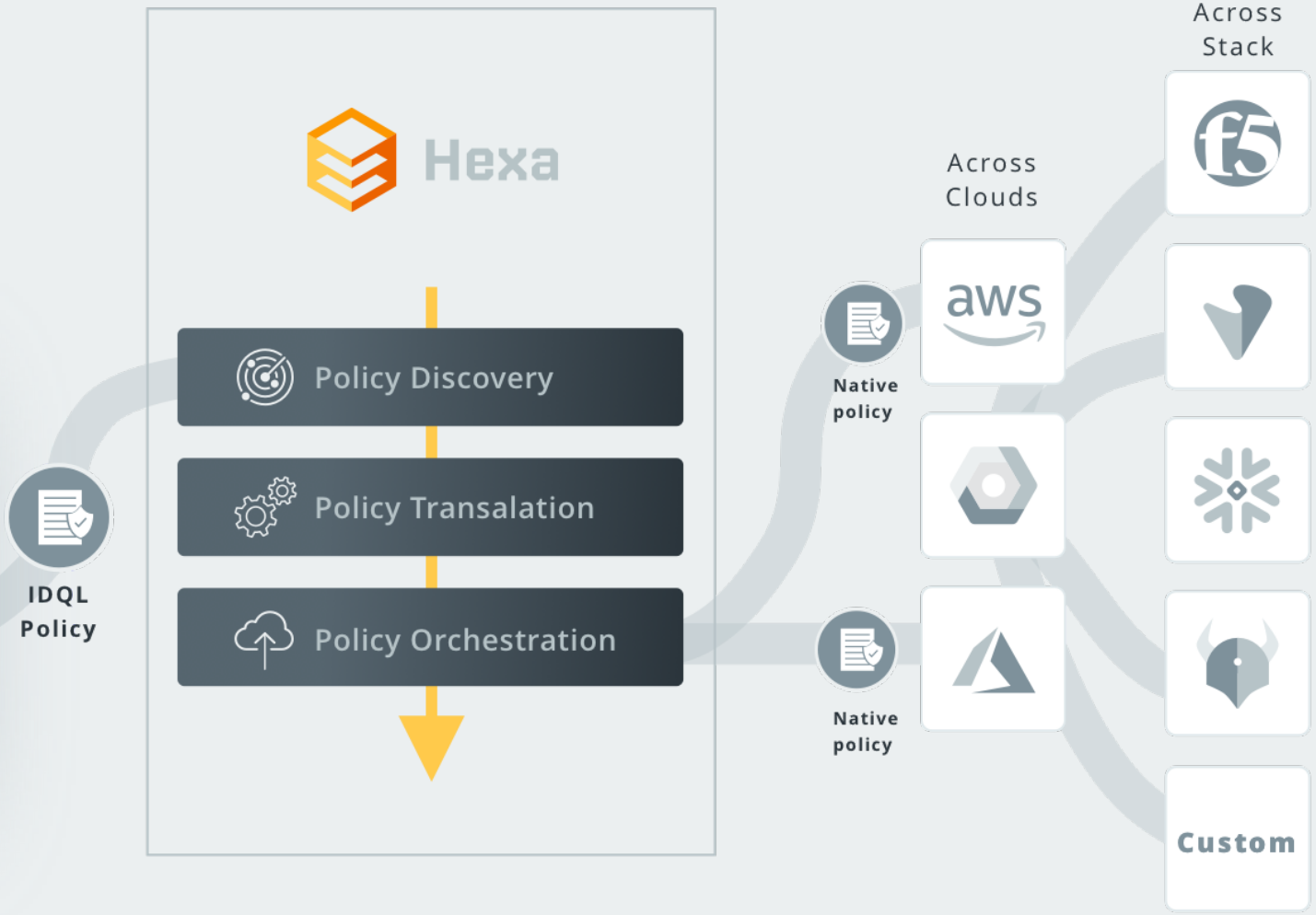
Authenticated Users	domain:initialcapacity.io domain:strata.io

With these actions

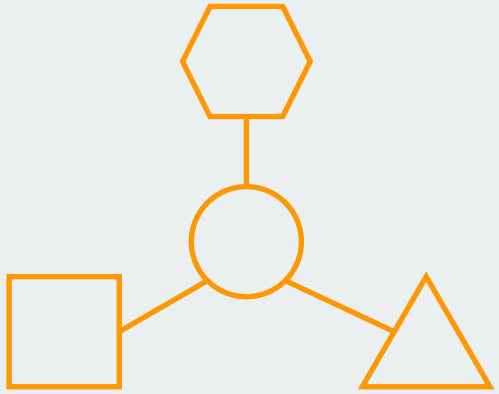
Action	HTTP Access

Object

Name	canary-bank-demo-42
Resource Type	APPLICATION
Cloud	Google cloud



IdentityQL (IDQL) is a new standard for access policy orchestration



IDQL

Hexa

IDQL Standardizes identity & access policy

- Universal policy model
- Define and distribute access policy
- Query and inspect distributed policy

IDQL is a new 'policy orchestration' standard

- Access policy across apps, platform, data and networks
- Built by community of leaders from enterprises and vendors

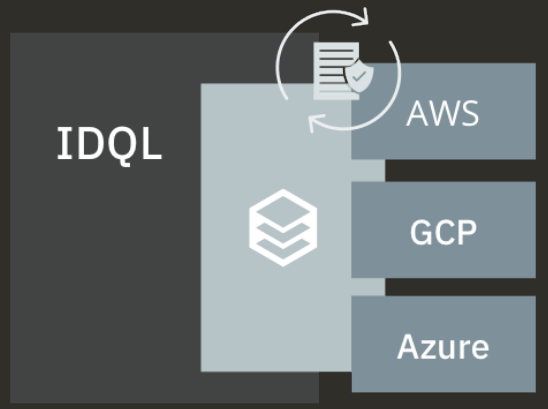
Hexa open source

- IDQL reference implementation
- Policy orchestration gateway
- CNCF (Apache license)

3 Policy Orchestration Use Cases

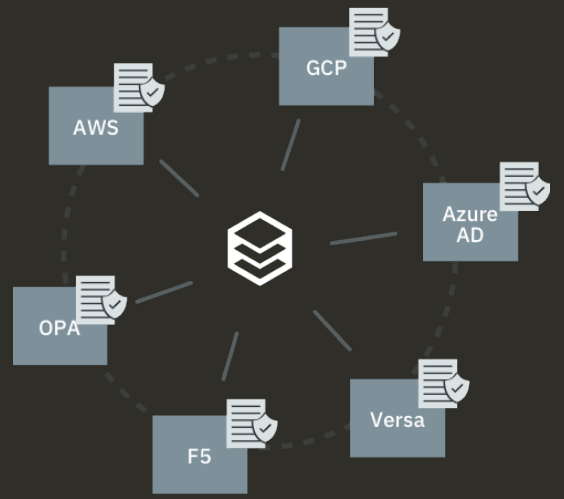
Policy Discovery

- Analyze your Identity Fabric and inventory key apps, data, and policies
- What apps exist? Where are they?
- What policies, users, and roles exist?



Policy Translation

- During policy discovery, translate native, imperative policies into IDQL
- During policy orchestration, translate IDQL into native, imperative policies of the target system(s)



Policy Orchestration

- Distribute policies to be enforced by IDPs, clouds, IaaS, and network systems
- Cloud-based architecture does not require proxy or local code
- Extensible, open source model to build your own integration

1. Discover Policies, Apps and Roles Across Clouds

Hexa

Consistent **inventory of apps and policies** across apps and clouds

Hexa

Policy Discovery

What apps are there?

What policies are there?

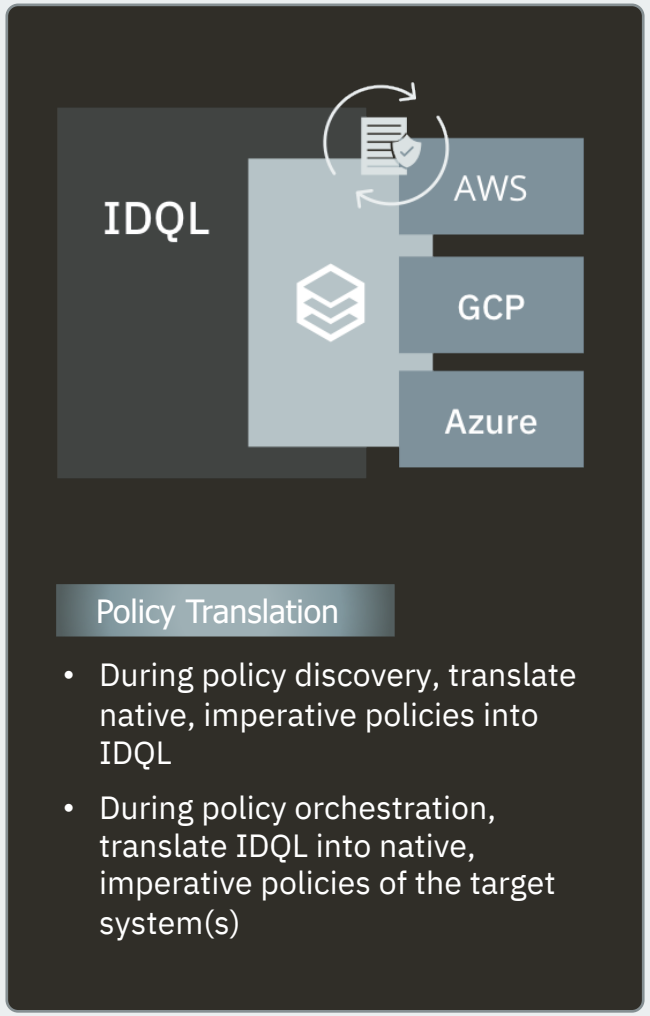
How are apps are being accessed?

What roles are used?

Hexa IDQL Policy Orchestration Connectors – Proxyless and Agentless

Okta Connector	Azure Connector	GCP Connector	AWS Connector	OPA Connector	SNOW Connector	F5 Connector	Custom Connector
Okta API	Azure API	GCP API	AWS API	REGO API	SNOW API	F5 API	Custom API
Okta	Azure	Google Cloud	AWS	OPA	Snowflake	Nginx	Custom

2. Translate Native Policies into IDQL




```
"applicationId": "8763f1c4-0000-0000-0000-158e9ef97d6a",
"appRoles": [
    {
      "allowedMemberTypes": [
        "User"
      ],
      "displayName": "editProfile",
      "id": "d1c2ade8-0000-0000-0000-6d06b947c66f",
      "isEnabled": true,
      "description": "Access policy enabling contract staff to edit profiles",
      "value": "accountEdit"
    }
  ],
"availableToOtherTenants": false,
{
    "customRules": [
      {
```


Hexa translates proprietary policies into open, declarative IDQL policies and into other native policies.

IDQL Policy Details v0.1

Subject

Authenticated Users	Type: OIDC ProviderID: corpOpenIdProvider Roles: employee

Action

Name	editProfile
actionUri	accountEdit

Object

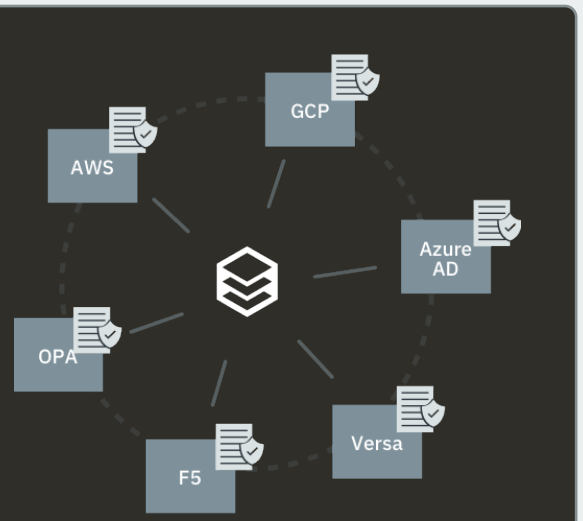
AssetId	hexaProfileService
pathSpec	/profile/*
Resource Type	APPLICATION
Cloud	Azure AD

Condition

Description	Employee access expires in 2025
rule	req.time lt "2025-01—01T00:00:00Z"
action	allow

3. Orchestrate policies across clouds and the stack

Hexa



Policy Orchestration

- Distribute policies to be enforced by IDPs, clouds, IaaS, and network systems
- Cloud-based architecture does not require proxy or local code
- Extensible, open source model to build your own integration

Propagate consistent policies across apps, data, platform and clouds

Simple and clear declarative policies

Hexa

Policy Orchestration

Access Policy: Trading App v2.31
Resource: Trading App
Allow: US & CAN Employees
Scope: US Data
Authentication: MFA required

Hexa IDQL Policy Orchestration Connectors – Proxyless and Agentless

Okta Connector	Azure Connector	GCP Connector	AWS Connector	OPA Connector	SNOW Connector	F5 Connector	Custom Connector
Okta API	Azure API	GCP API	AWS API	REGO API	SNOW API	F5 API	Custom API
Okta	Azure	Google Cloud	AWS	OPA	Snowflake	Nginx	Custom

Targets don't require coding

Hexa Policy Orchestration

- **Open Source**

  The Hexa Policy Orchestration Gateway will be freely available open source – Apache licensed.

- **Zero Overhead**

  Policy orchestration is done through admin processes, not runtime.

- **Simplified compliance**

  Easy to read access and user reports generated from across clouds.

- **East <-> West policy**

  Policy is unified across AWS, Azure and GCP (and any other cloud platform).

- **Enterprise-ready**

  Hexa works with your containers, Kubernetes, OPA, and CI/CD tools and processes.

- **Seamless API integration**

  Native integration with cloud and identity APIs insulates you from changing cloud APIs.

- **North <-> South policy**

  Policy is also managed across the stack, across apps, data, platform and networking.

- **Simplified security**

  IDQL is a clear, declarative policy that is simple to understand and enforce.

- **Natively integrated with other CNCF tools**

  Orchestrate OPA policies with your other access management systems, orchestrate K8s policies and run Hexa on Kubernetes.

Benefits of Policy Orchestration with Hexa and IDQL

Hexa

- **Agentless and proxyless Policy Orchestration**

  Implement in minutes without changes to your infrastructure.

- **Distributed policy management**

  Orchestrate access policy securely through APIs with no change required to target systems.

- **Universal Access Policy**

  Manage access policy that works across disparate systems to help enable a Zero Trust Architecture.

- **Policy-as-Code**

  Bring identity and access policy into code for large-scale automation.

- **Declarative Policy**

  Understand who has access to which apps and data at a glance.

- **Break lock-in**

  Break lock-in and enjoy portability and vendor choice

IdentityQL - IDQL

Hexa

Why does the world need another standard?

- Distributed architectures require policy consistency across disparate platforms, domains and technologies.
- Without a standard there will be inconsistency, risk through silos, greater cost and lock-in.
- Identity and Access Policy has not been standardized yet and has kept IAM fragmented.

What problems does IDQL solve?

- IDQL solves the definition, enforcement and management of consistent user identity and access policies across distributed systems. Both on the East West and North South axis.
- IDQL brings X-as-Code to User Identity and Access Policy to be easily consumed in modern DevOps environments.

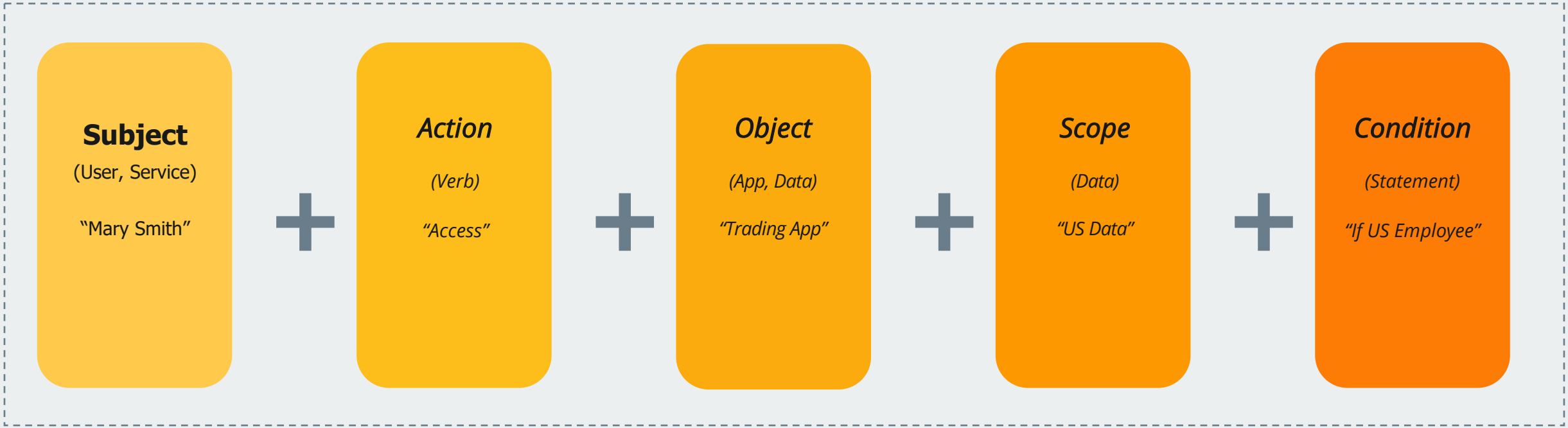
How does IDQL compare to other standards?

- OPA (Open Policy Agent (CNCF) – Focused on K8S cluster management, networking, and microservices - not policy orchestration.
- SPIFFE/Spire (CNCF) – focused on App to App identity using x509, not end user identity.
- SAML, OIDC, OAuth are all protocols for SSO and delegated authorization but not end user identity policy.
- SCIM is concerned with user provisioning but does not address how policies are defined and orchestrated.
- XACML (OASIS) – focused on fine grained authorization at runtime and not end user identity policy orchestration.

IDQL – Simple, Declarative Access Policies

Hexa

Context



Simple, **declarative** syntax is easy to read and understand by people.
Clearer understanding of policies reduces risk

How to make IDQL a standard?

Leverage experience co-authoring SAML which has become the gold standard in identity
Define spec and get buy-in, then bring into standards body for standardization.

How We Are Building IDQL

- Working with key customer segments
  - Financial, Insurance, Retail, etc.
- Engaging industry experts (SAML contributors, Industry Analysts & partners)
- Cloud Native Computing Foundation to manage the open standard
  - IDQL spec
  - Open source code

IDQL Working Group

- Policy specification and API definition

IDQL Focus Groups

- Application module
- Platform module
- Data module
- Network module

Building IDQL With The Open Community

Hexa

Authors

Contributors

Reviewers

Adopters

Supporters

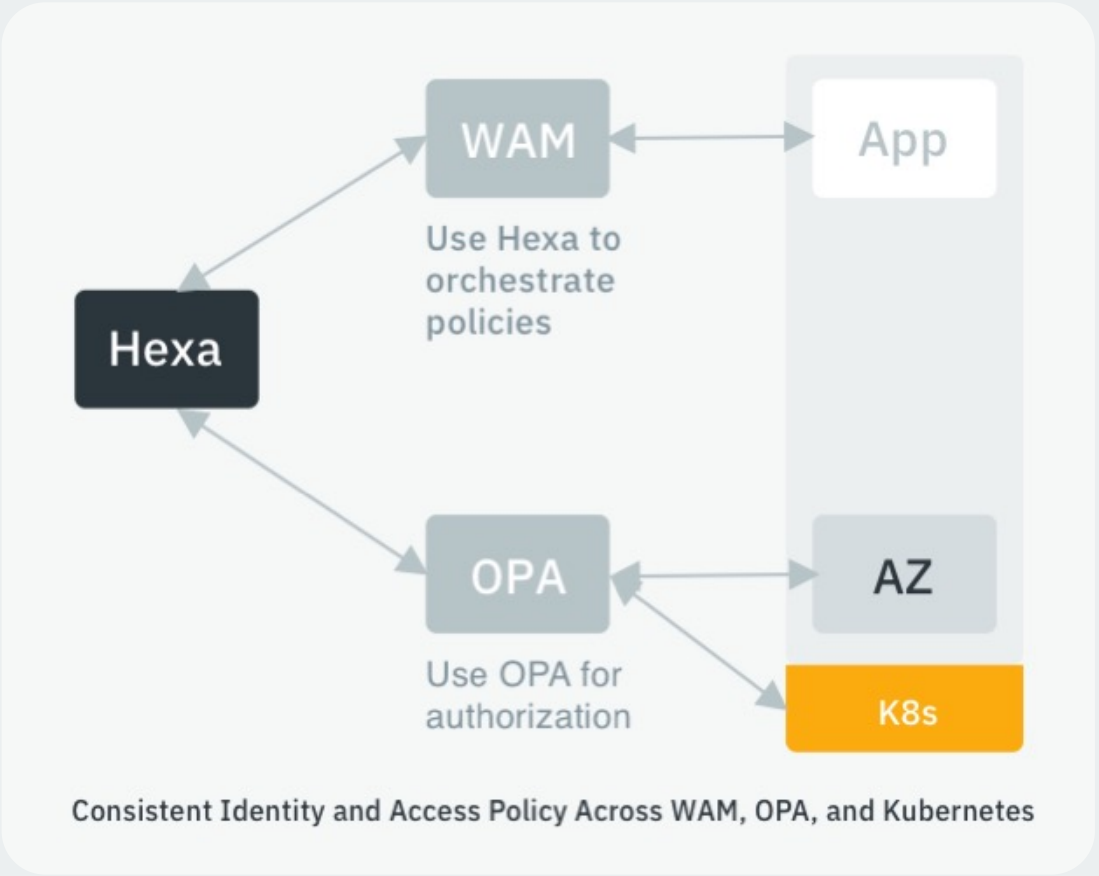
- **Authors**: Edit the policy and API specifications
- **Contributors**: Provide thought leadership and design, use cases, design and environmental requirements
- **Reviewers**: Design reviews, code reviews
- **Adopters**: Use Hexa and IDQL in Pilot and Production
- **Supporters**: Members of the Hexa and IDQL Community

Use Hexa to orchestrate OPA policies

Hexa is a Policy Management Point (PMP)

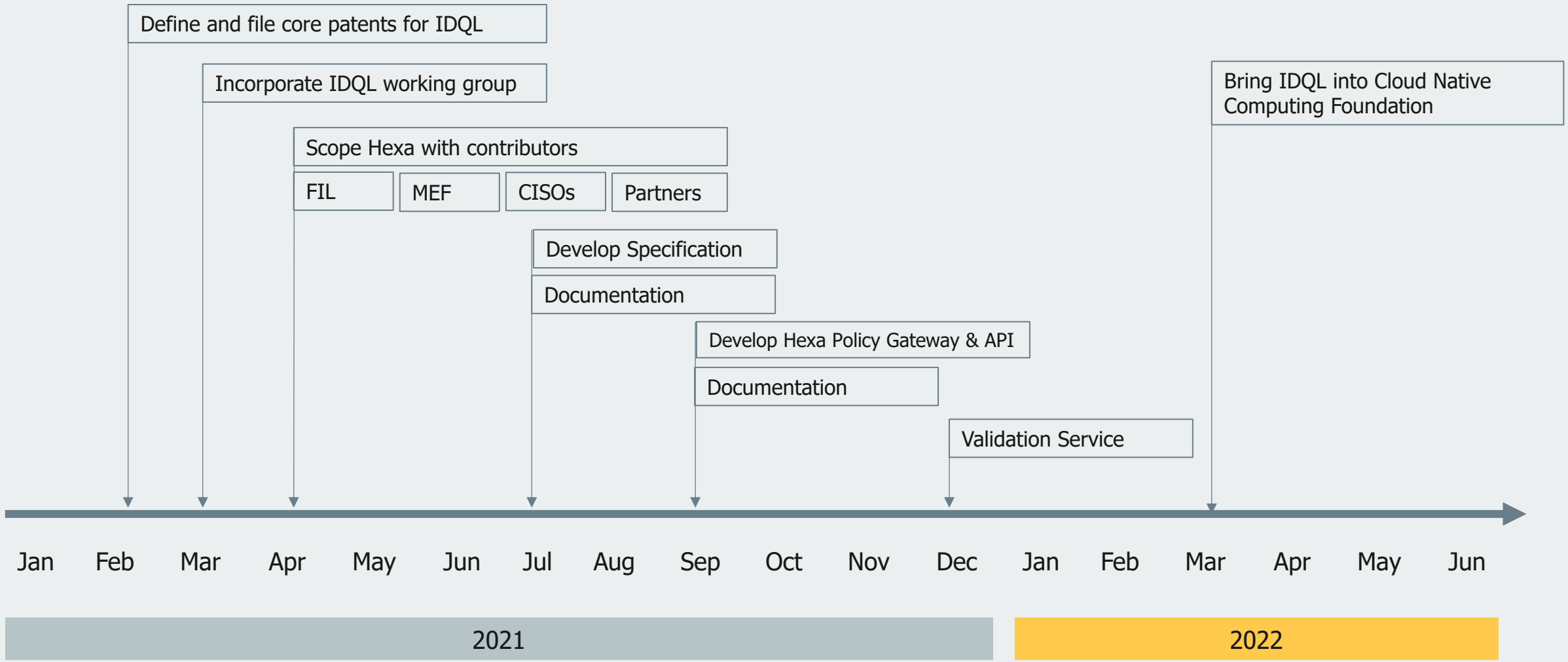
OPA is a Policy Enforcement Point (PEP)

Use Hexa to align access policies across web apps, authorization and Kubernetes



Consistent Identity and Access Policy Across WAM, OPA, and Kubernetes

Working Timeline



IDQL Working Group Kick Off – June 2021