Managing Large Database Landscapes in Multi-Cloud DBaaS Environments

Zürcher Kantonalbank

Swiss PGDay
27. Juni 2024

Andreas Geppert

About Me

- Diploma in Computer Science from the Karlsruhe Institute of Technology
- Dr. in Computer Science from the University of Zurich
- Visiting Scientist at IBM Almaden Research Center
- Platform Archtitect at Credit Suisse (14 years)
- Database Architect, Consultant and Engineer at Swiss Re (6 years)
- **DevOps Solution Engineer at ZKB (since July 2022)**

- Vice president of the Swiss Postgres Users Group

- https://www.linkedin.com/in/andreas-geppert-6964a0178/

Content

Zürcher Kantonalbank

- Requirements
- Service Classes and Catalogs
- Abstraction to enable Multi-DBMS
- DBaaS-Architecture
- Sharing vs. Dedicated
- Self-Service
- Data-driven Platform Management

Current Trend

- Most large enterprises move to the cloud
- Postgres is the perfect choice as strategic and standard database management system in the cloud
- How to architect, build, and run your database landscape with Postgres and possibly other DBMSs in the clouds?

Requirements

Zürcher Kantonalbank

- Security and Hardening
- Multi-Cloud
- Multi-Product
- Multi-Client
- Resilience
- Autonomy and agility
- Standardization, Automation, and Self-Service
- Integration in the enterprise eco system and processes
- Data-driven management
- Cost effectiveness

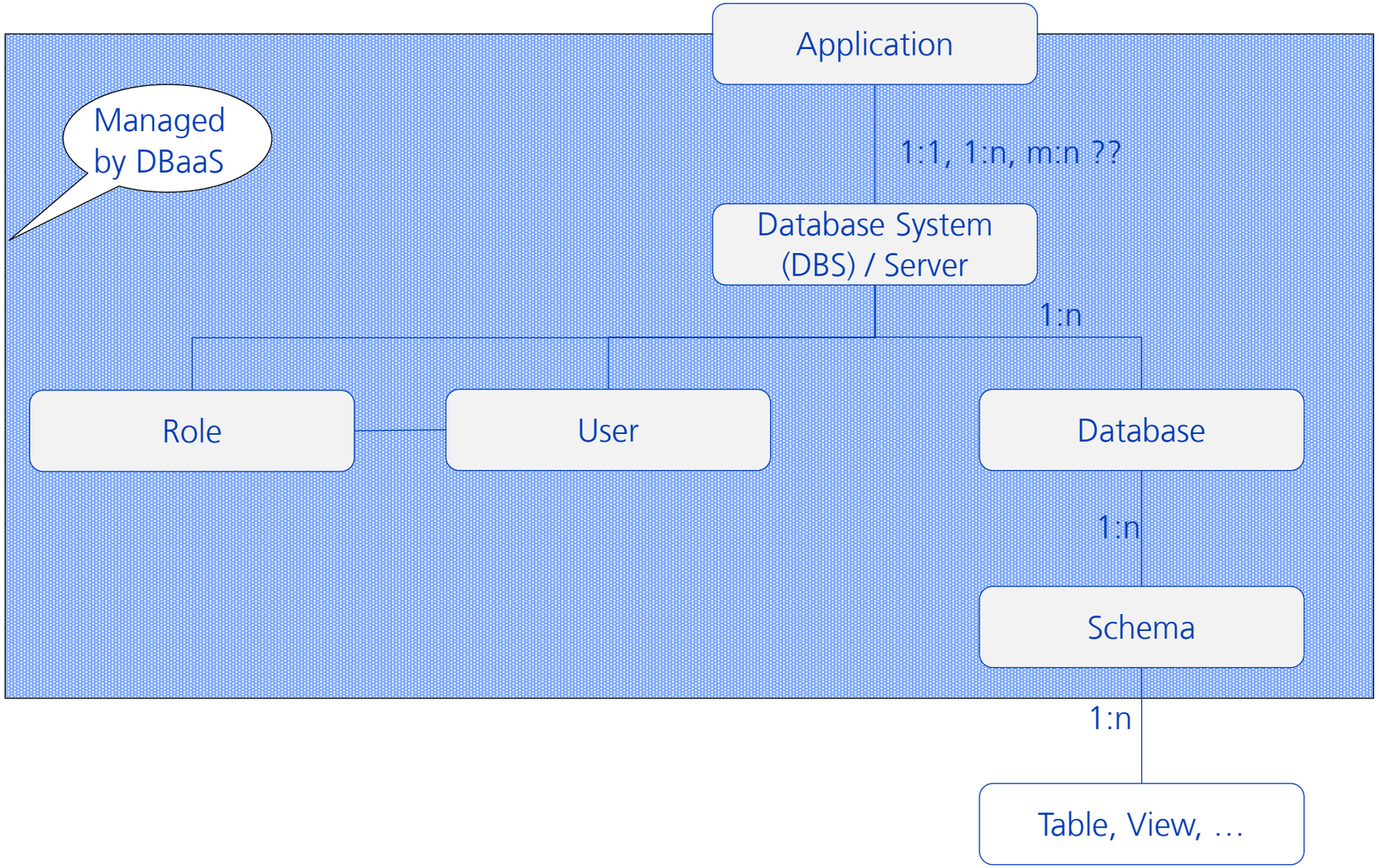
DBaaS Service Catalogs
Meeting a spectrum of application requirements

Zürcher Kantonalbank

		D (Basic)	C (Standard)	B (Premium)	A (Platinum)
	DBMS	Postgres			
DR	DR	No	Yes	Yes	Yes
	RTO	-	24h	4h	4h
	RPO	-	24h	0h	0h*
Availability	HA	No	No	Yes	Yes
	Planned Downtime	N/A; depends on provider			
Performance	Scalability	No	No	Vertical	V/H
	Isolation	No	No	Partially	Yes

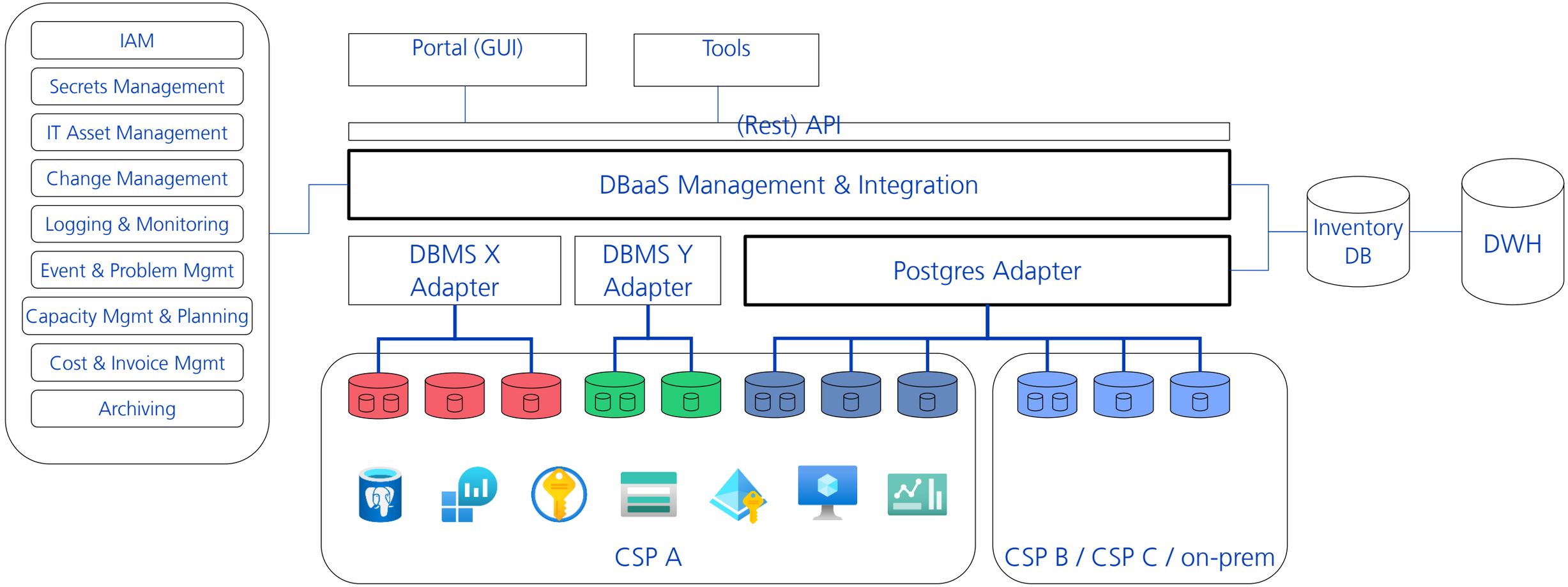
Abstractions

- Abstraction helps to address common issues once and for all DBMSs
- Particularly integration is DBMS-independent



DBaaS Architecture

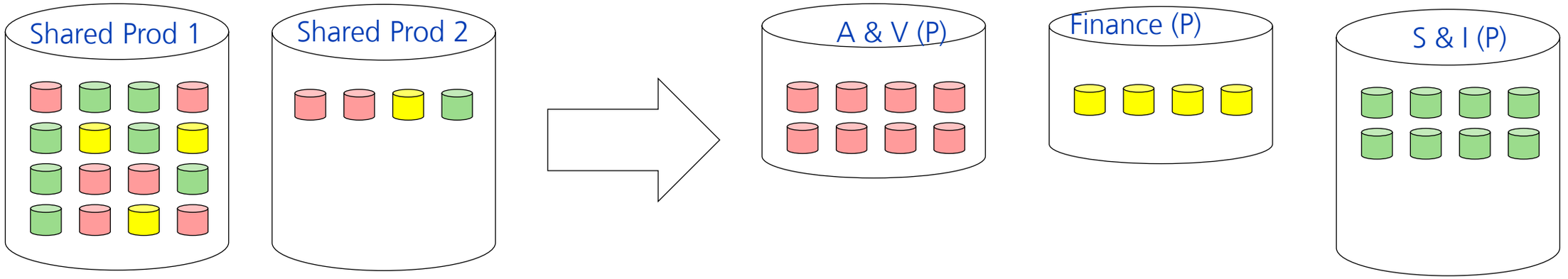
Zürcher Kantonalbank



Sharing vs Dedicated

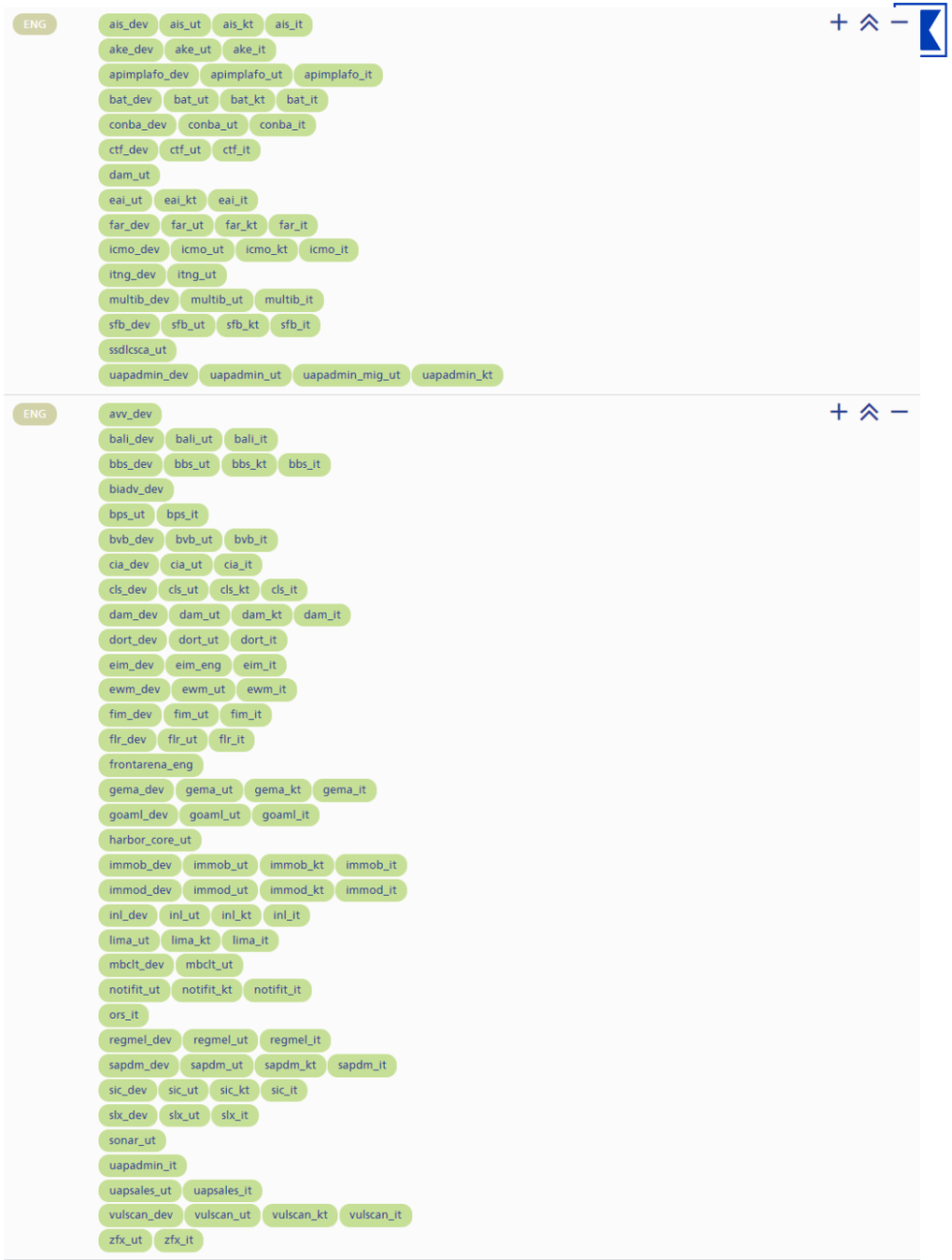
Zürcher Kantonalbank

- Many applications do not utilize an own server
- Trade-off:
  - Sharing a server between multiple applications then saves cost and optimizes utilization
  - But applications can impact each other
- Let domains and applications decide autonomously
- Default: place servers and databases along the boundaries of domains/portfolios



Sharing vs Dedicated
Current Situation (Partial Snapshot)

- Currently, almost 300 databases
- In ~ 30 Postgres servers (shared and dedicated)
- Upcoming re-placement along portfolio boundaries upcoming, with upgrade to version 16



DBaaS Self-Service

Zürcher Kantonalbank

- Application autonomy and agility require self-service for users
- All routine tasks should be available as self-service
- REST API implements self-service features
- Adequate interfaces should be provided on top of REST API (GUI/portal)

- CRUD Database System
- CRUD Database
- CRUD Schema
- CRUD User
- CRUD Role
- CRUD Role Membership
- Export/import
- Backup & Archiving
- SQL Execution
  - DDL
  - DML
- Migration

DBaaS REST API
Initial Pilot

Zürcher Kantonalbank

infra.ApplicationIfc

infra.DbmsVersionIfc

infra.ServerIfc

infra.DatabaseIfc

GET `/database/list/all` returns all DBs.

GET `/database/listapp/{name}` returns all DBs by name.

POST `/database/add/{server}/{name}` Adds a Postgres logical db to Server.

infra.DbUserIfc

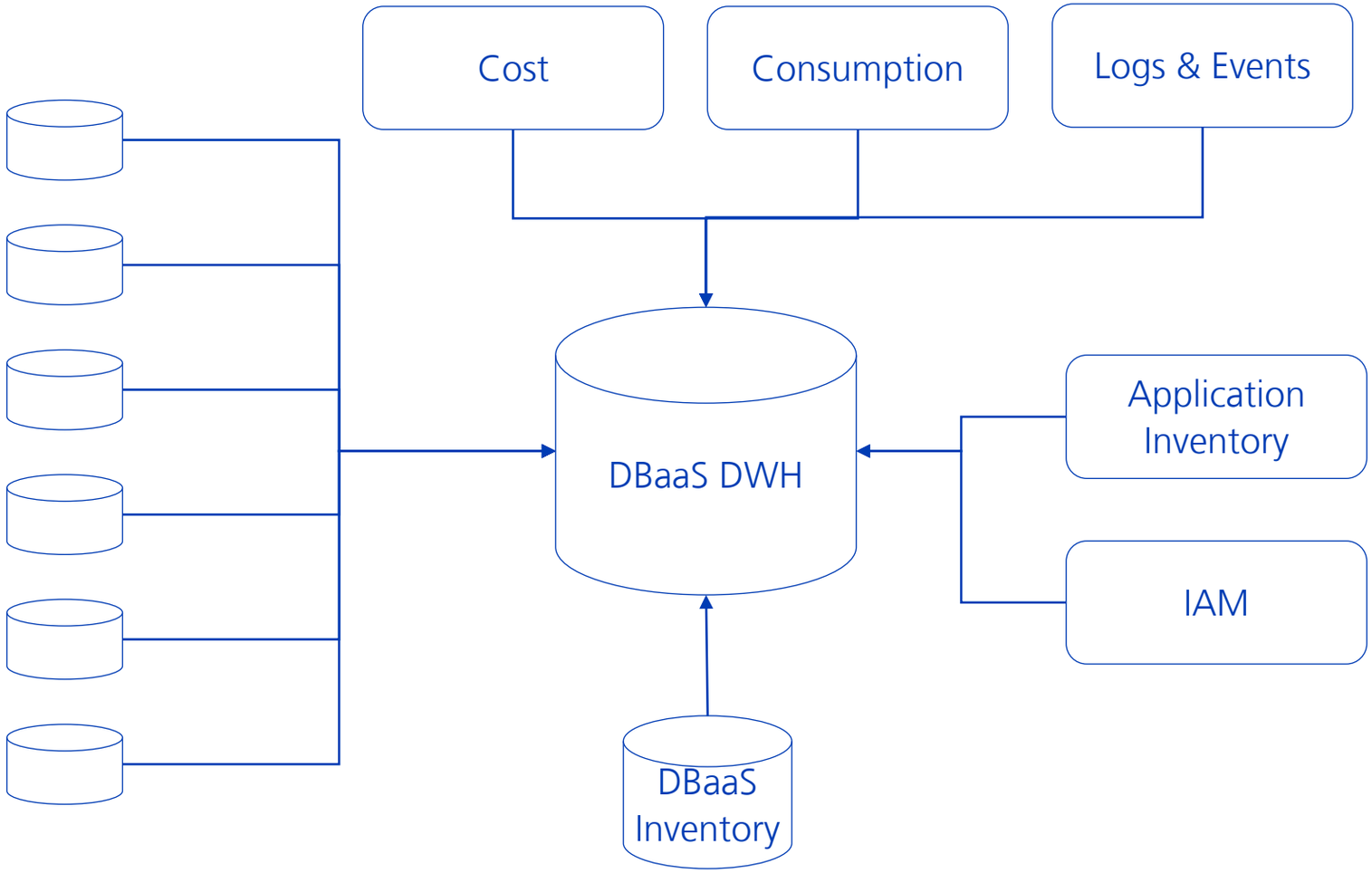
infra.DbRoleIfc

infra.DbSchemaIfc

The DBaaS Data Warehouse

Zürcher Kantonalbank

- The DBaaS data warehouse collects historical data from all relevant sources and enables data-driven platform management, such as:
  - Usage reporting and analysis
  - Invoice verification and cost allocation
  - Security monitoring and configuration drift detection



The DBaaS Data Warehouse

- Sample dashboards showing number of applications per portfolio/domain and environments, and databases by environment

Open Issues / Room for Improvement

Zürcher Kantonalbank

- Postgres User Management requires updates to the pg_hba.conf
  - See Automatisierte Benutzer- und Zugangsverwaltung in DBaaS-Umgebungen @ PGConf.de 2022
- Built-in Transparent Data Encryption
- Resource manager (as in other DBMS) to limit resource consumption by individual databases in shared servers
  - Database connection limit
  - Monitoring + «Eviction» of unfriendly neighbors into own servers

Conclusion

- Automation for Postgres in Azure completed
- Currently completing the first version of the DBaaS self-service
- Additional DBMSs, further clouds?
- Cross-platform setups

?!?