FIVE RULES OF MICROSERVICES

Armagan Amcalar

I T.A.K.E. Unconference

April 7th, 2020

Who am I?

Armagan Amcalar

Software architect, leader, mentor, speaker

Founder @ Lonca Works

dashersw dashersw

AUTHORED ON GITHUB

erste

cote

dovecote

semaver

vieux

brain-bits

pedalboard.js

tartJS

geneJS

jira-bot

Outline

What is and is not a Microservice

An example e-commerce application

5 rules of microservices

Contemporary tools to enable true microservices

All microservices are distributed applications.

Not every distributed application is a microservices application.

What is not a microservice?

Async operations with queue (and other) systems

Having 50 different queue consumers (for notifications, logging, reconciliation services, e-mailing etc) doesn't mean you have 50 microservices

Multiple programs running on a single machine and communicating over HTTP

What is a Microservice?

If you are breaking down the fulfillment of a client request into multiple collaborating services that run in their own memory space, then you are doing microservices.

THE BASELINE

dashersw

Microservices

The quality of your approach depends on how well you apply Domain Driven Design and other best practices.

You have to —

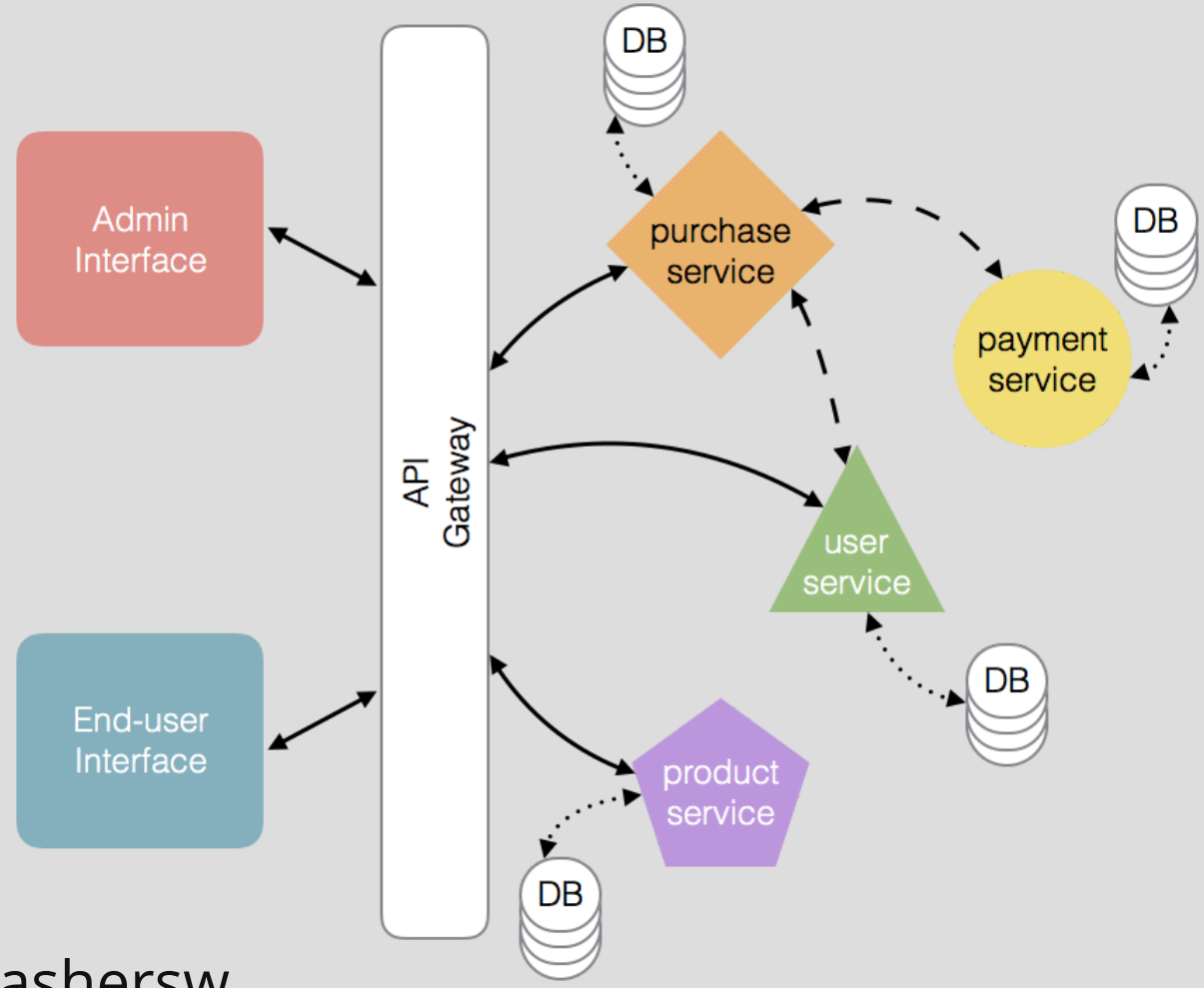
Embrace change

Give autonomy to teams, to services

Automate testing, configuration, deployment, monitoring and more

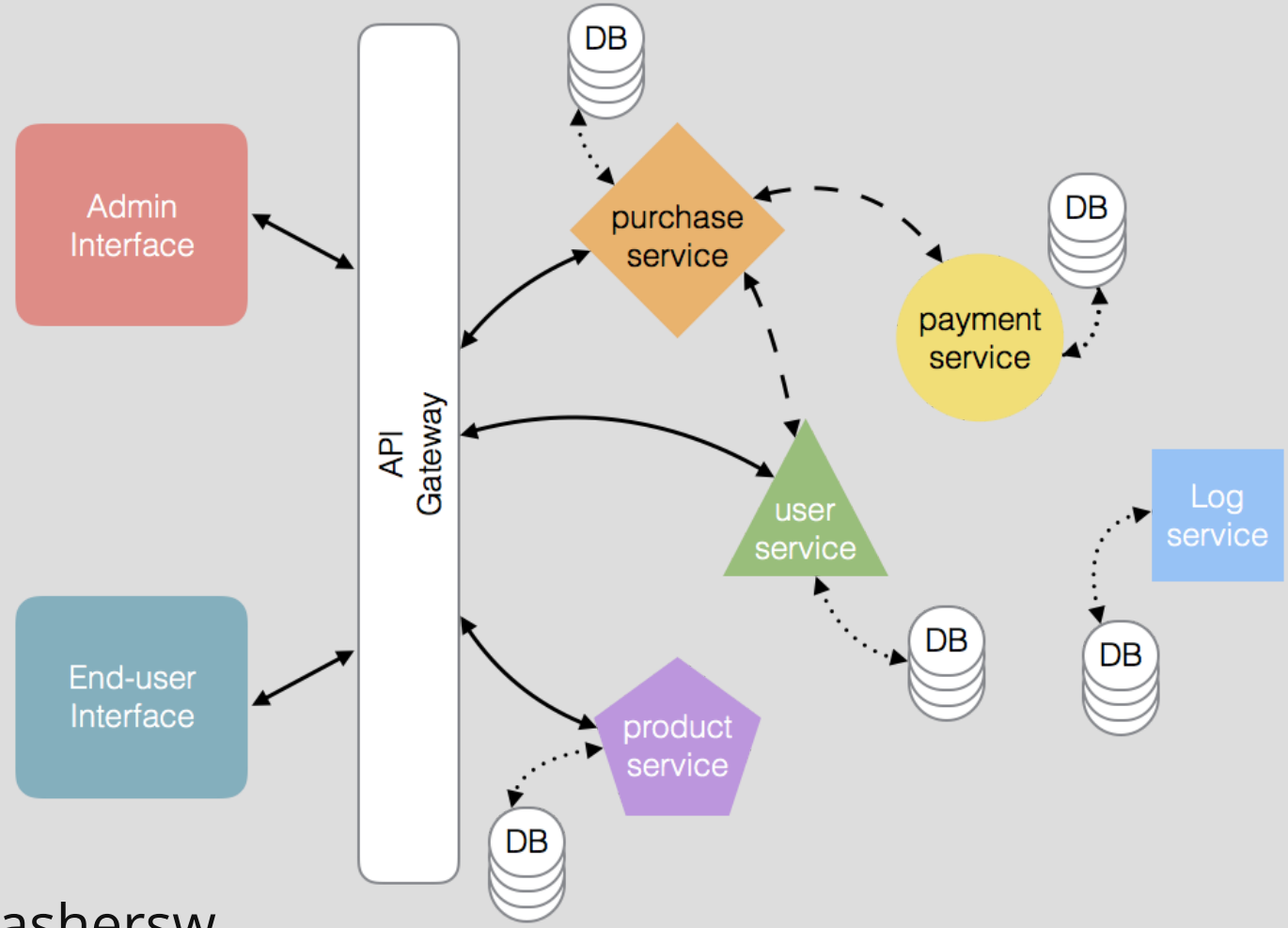
Favor cattle over pets

An example e-commerce app



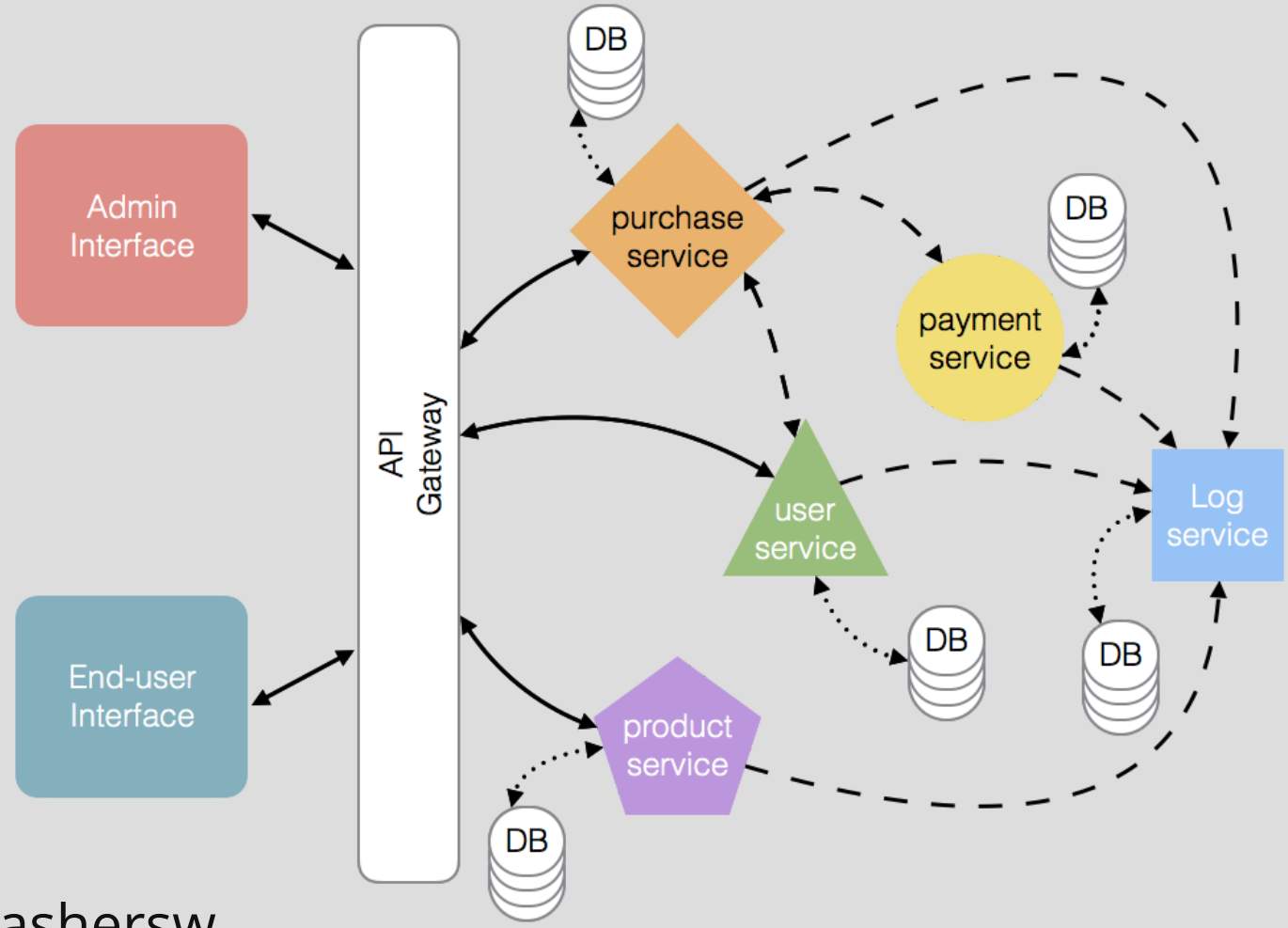
dashersw

Rule #1: Auto-discovery



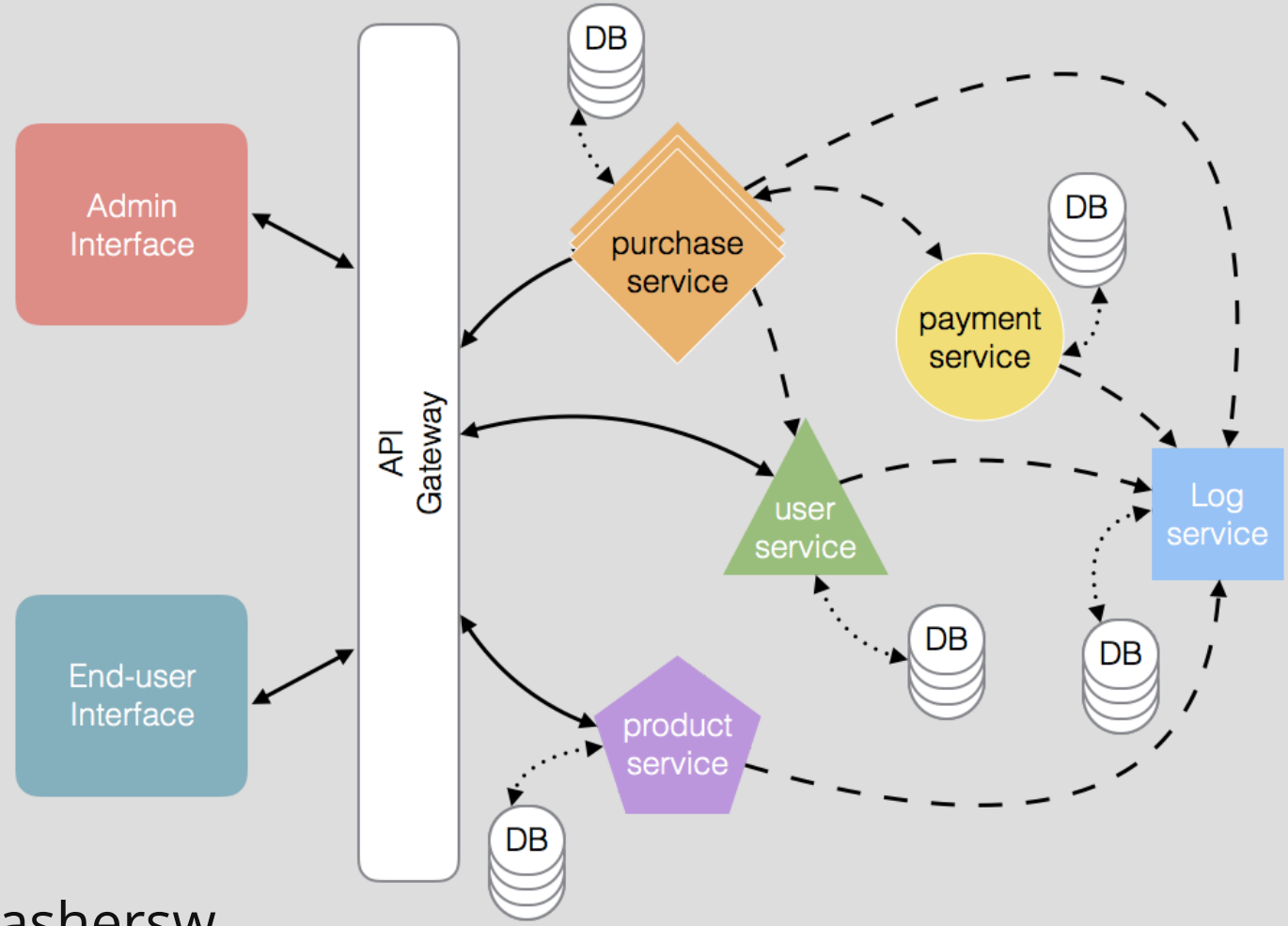
dashersw

Auto-discovery



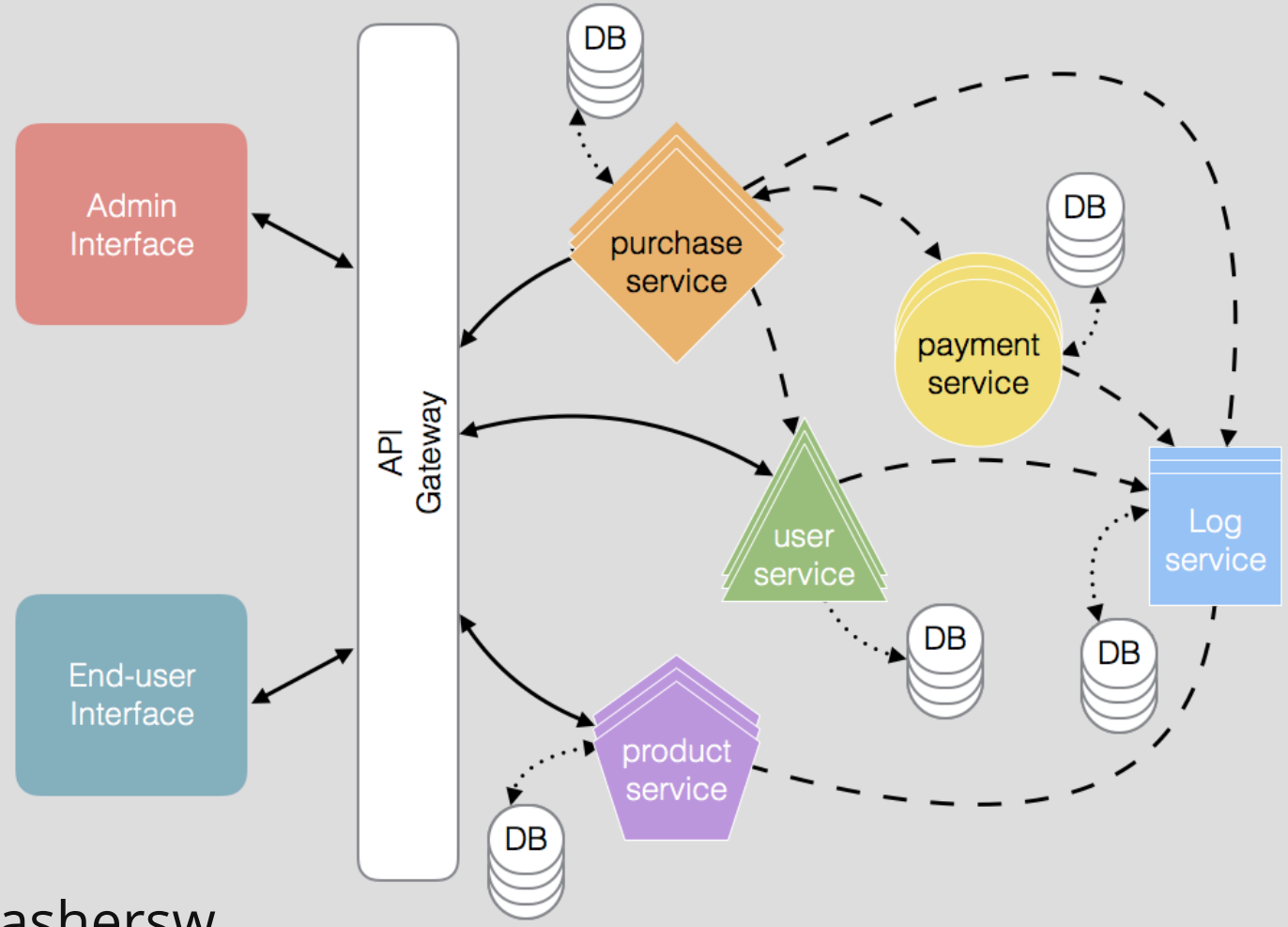
dashersw

Rule #2: Zero-configuration



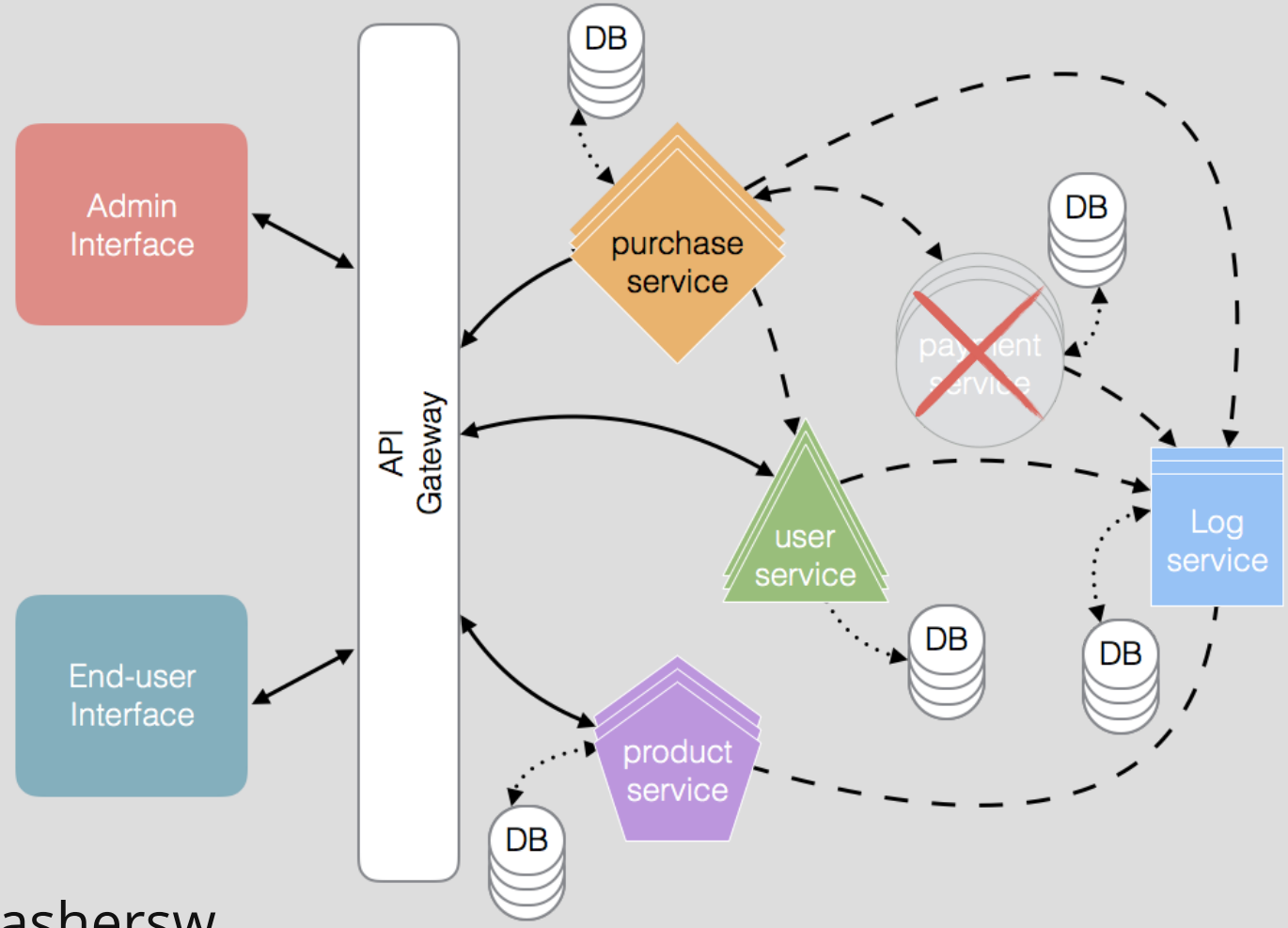
dashersw

Rule #3: Highly-redundant



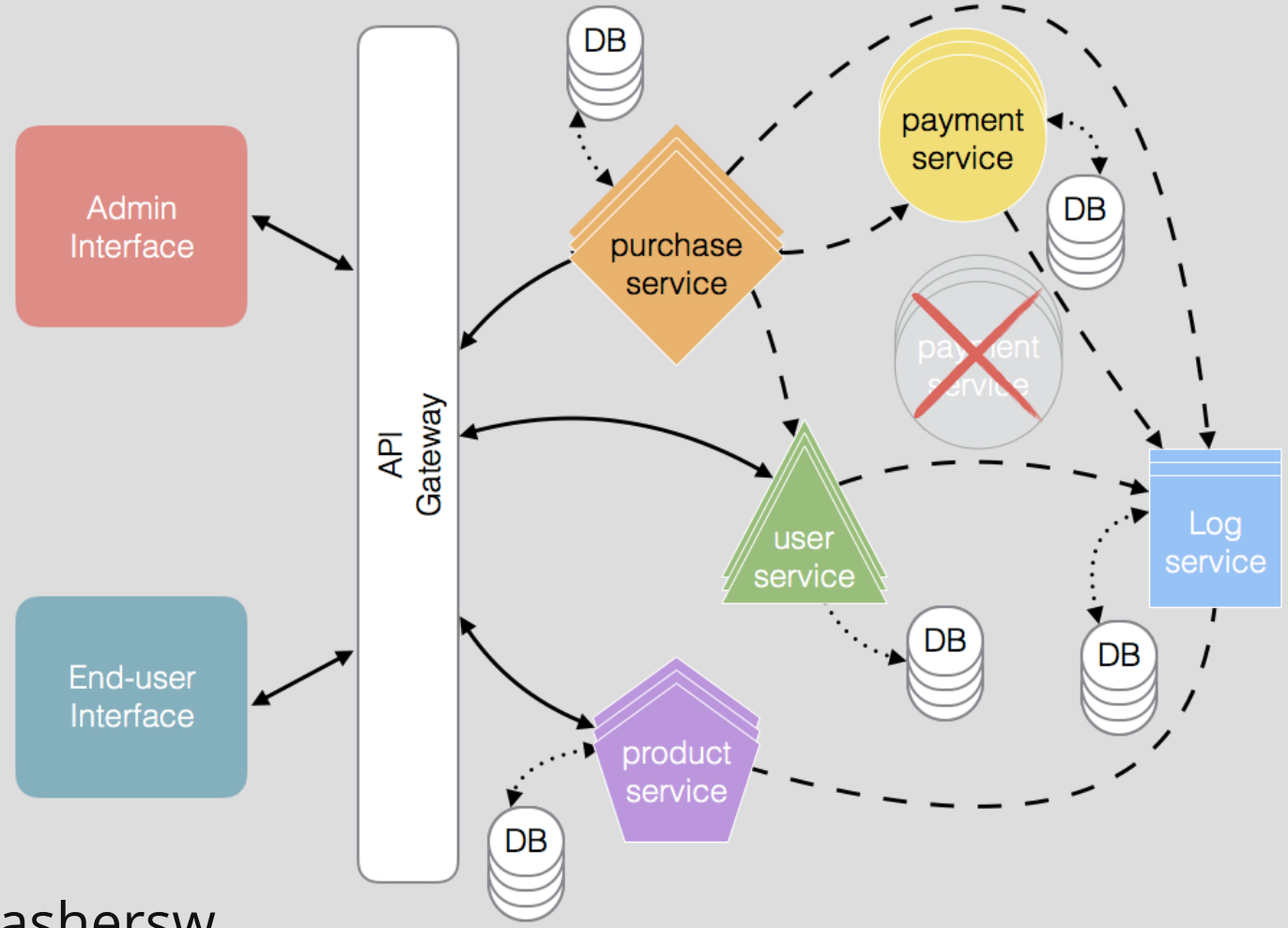
dashersw

Rule #4: Fault-tolerant



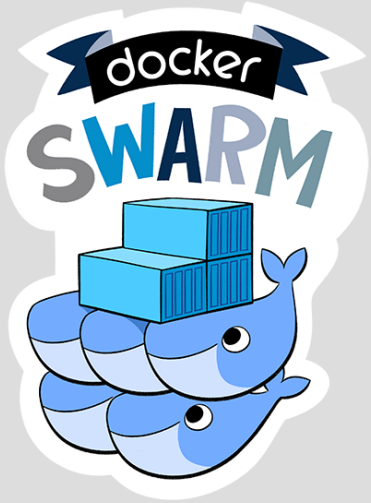
dashersw

Rule #5: Self-healing



dashersw

Groundwork: Orchestration Services



dashersw

Groundwork: Orchestration Services

Enable auto-discovery,
roll outs / roll backs,
self-healing,
secret & configuration management

Rule #1: auto-discovery
Rule #2: zero-configuration
Rule #3: Highly-redundant
Rule #5: Self-healing

dashersw

What about Rule #4: Fault-tolerance?

What about Rule #4: Fault-tolerance?

Circuit-breakers

Retries

Timeouts

What about Rule #4: Fault-tolerance?

Circuit-breakers

Retries

Timeouts

dashersw

What we have left out

Other common concerns for modern apps

Observability

(Distributed) Tracing

Authentication

Authorization

Secrets management

Volume management

All microservices are distributed applications.

Not every distributed application is a microservices application.

thank you!

slides:

https://slides.com/armaganamcalar/five-rules-of-microservices

Let's keep in touch!

Armagan Amcalar

dashersw