Keeping Things Running

Michael L Perry
SOFTWARE MATHEMATICIAN

@michaellperry immutablearchitecture.com

Bootstrap teams

Define standards

Applications are
independent and integrated

Teams need
autonomy and standards

Members have
common goal and specialized skills

Team Roles

System Operators

Dependencies
Tracing

Database Administrators

Entities
Integrity
Performance

Developers

Delivery
Documentation
Context

Testers

Behavior
Synthesis

Operational Documentation

Developing

Running

Mindset

Development	Operations
Requirements	Load
Algorithms	Resource Utilization
Behaviors	Anomalies
Happy path	Sad path
Exceptions are in context	Exceptions are missing context

Runbook

- Components
- Dependencies
- Communication channels
- Behaviors
- Mitigations
- Living document

Application Database Documentation

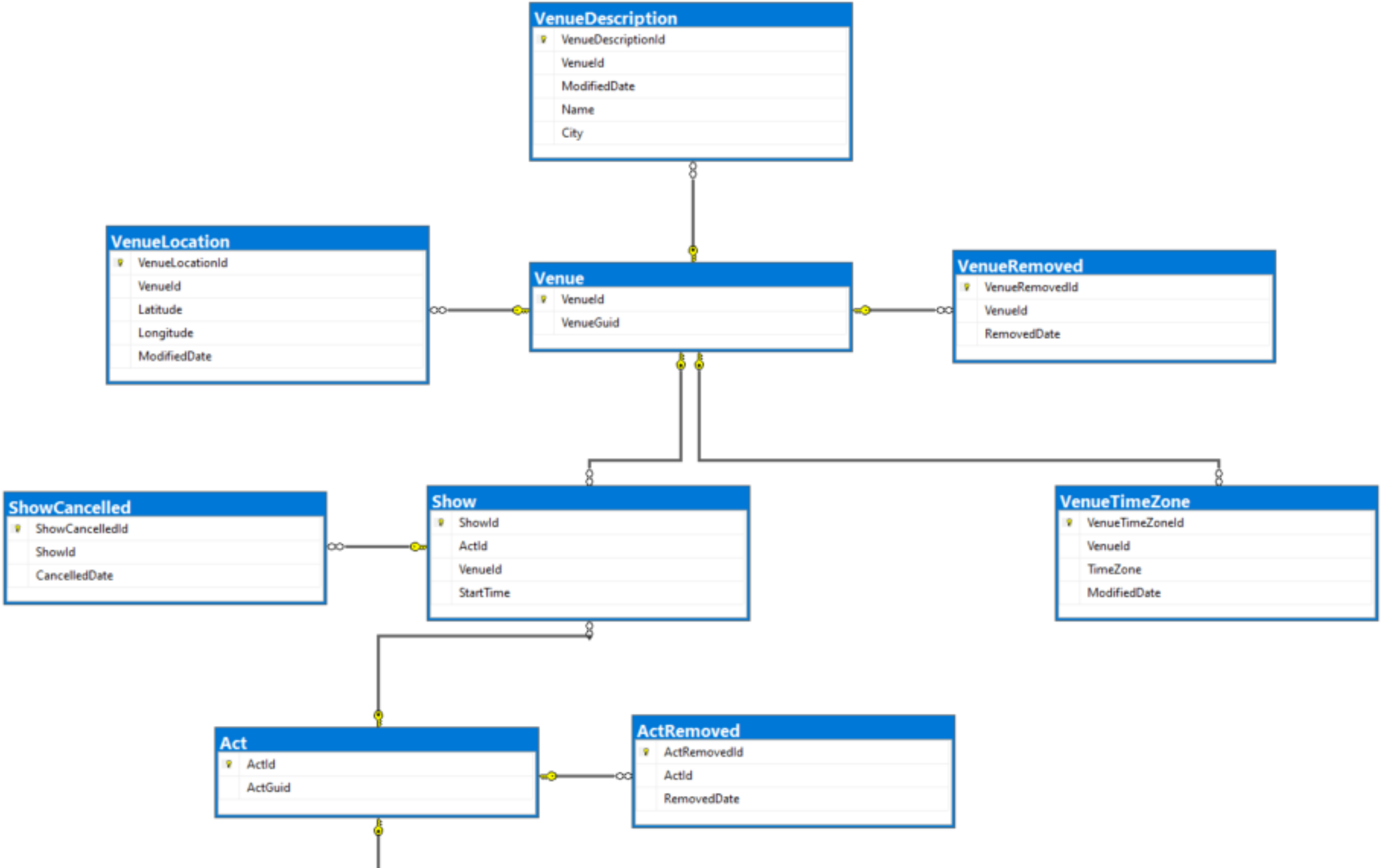
One database per application

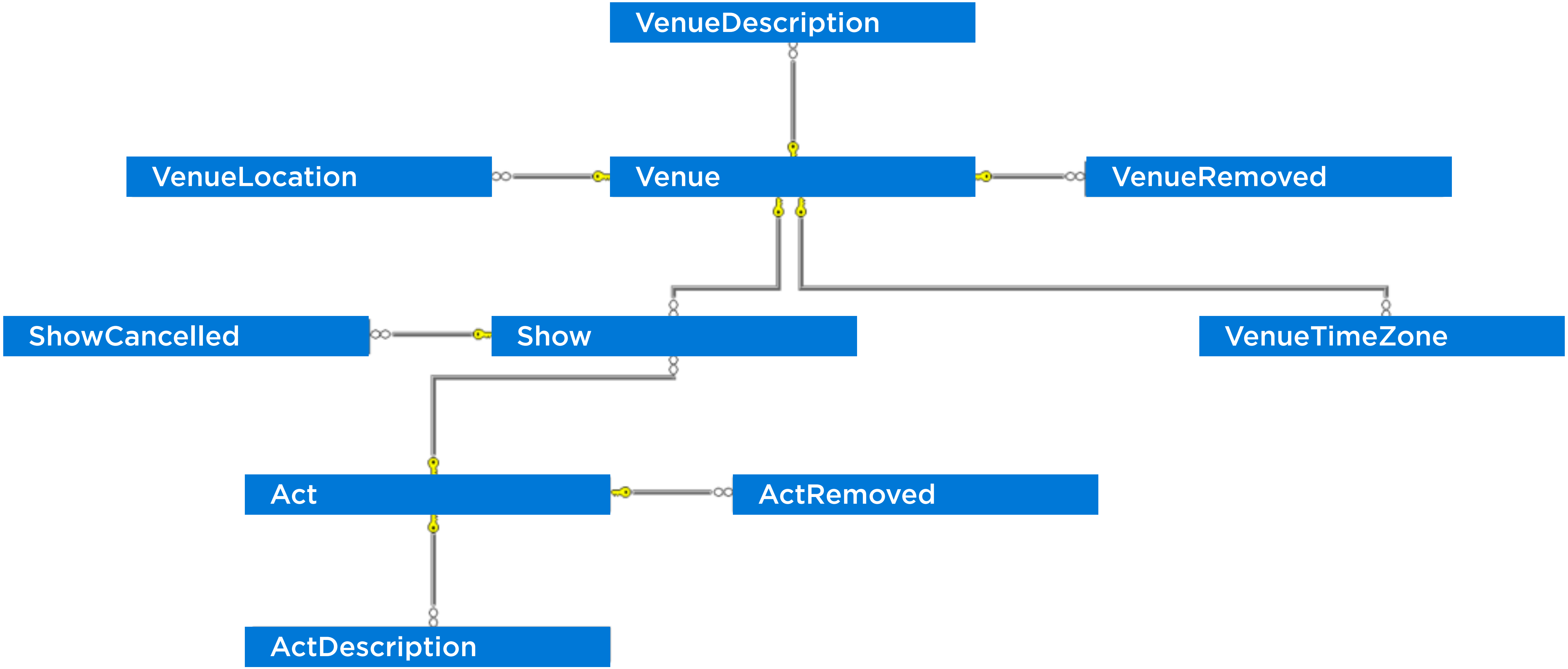
Operational DBA

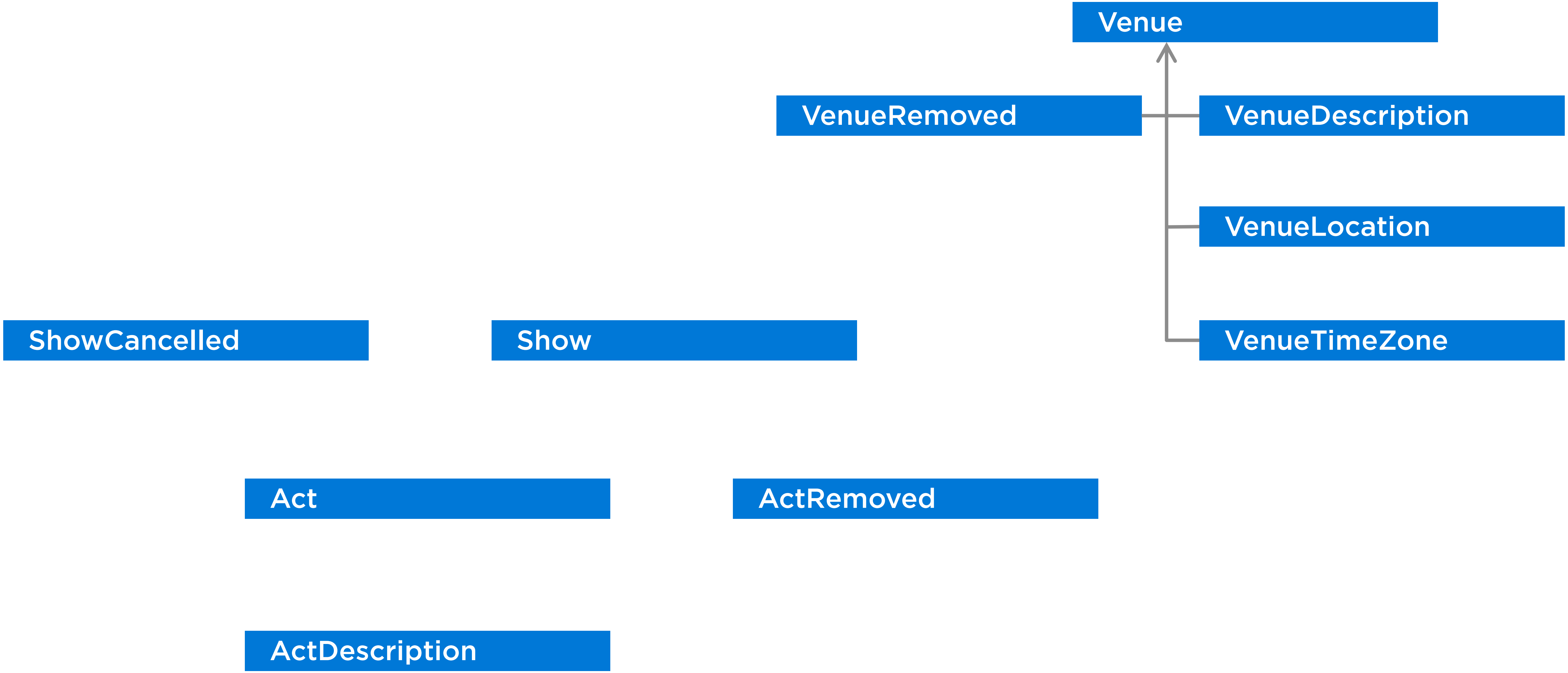
Health & integrity

Long-running query

Large table


VenueDescription
VenueDescriptionId
VenueId
ModifiedDate
Name
City
VenueLocation
VenueLocationId
VenueId
Latitude
Longitude
ModifiedDate
Venue
VenueId
VenueGuid
VenueRemoved
VenueRemovedId
VenueId
RemovedDate
ShowCancelled
ShowCancelledId
ShowId
CancelledDate
Show
ShowId
ActId
VenueId
StartTime
VenueTimeZone
VenueTimeZoneId
VenueId
TimeZone
ModifiedDate
Act
ActId
ActGuid
ActRemoved
ActRemovedId
ActId
RemovedDate


VenueDescription
VenueLocation
Venue
VenueRemoved
ShowCancelled
Show
VenueTimeZone
Act
ActRemoved
ActDescription



Venue

VenueRemoved

VenueDescription

VenueLocation

ShowCancelled

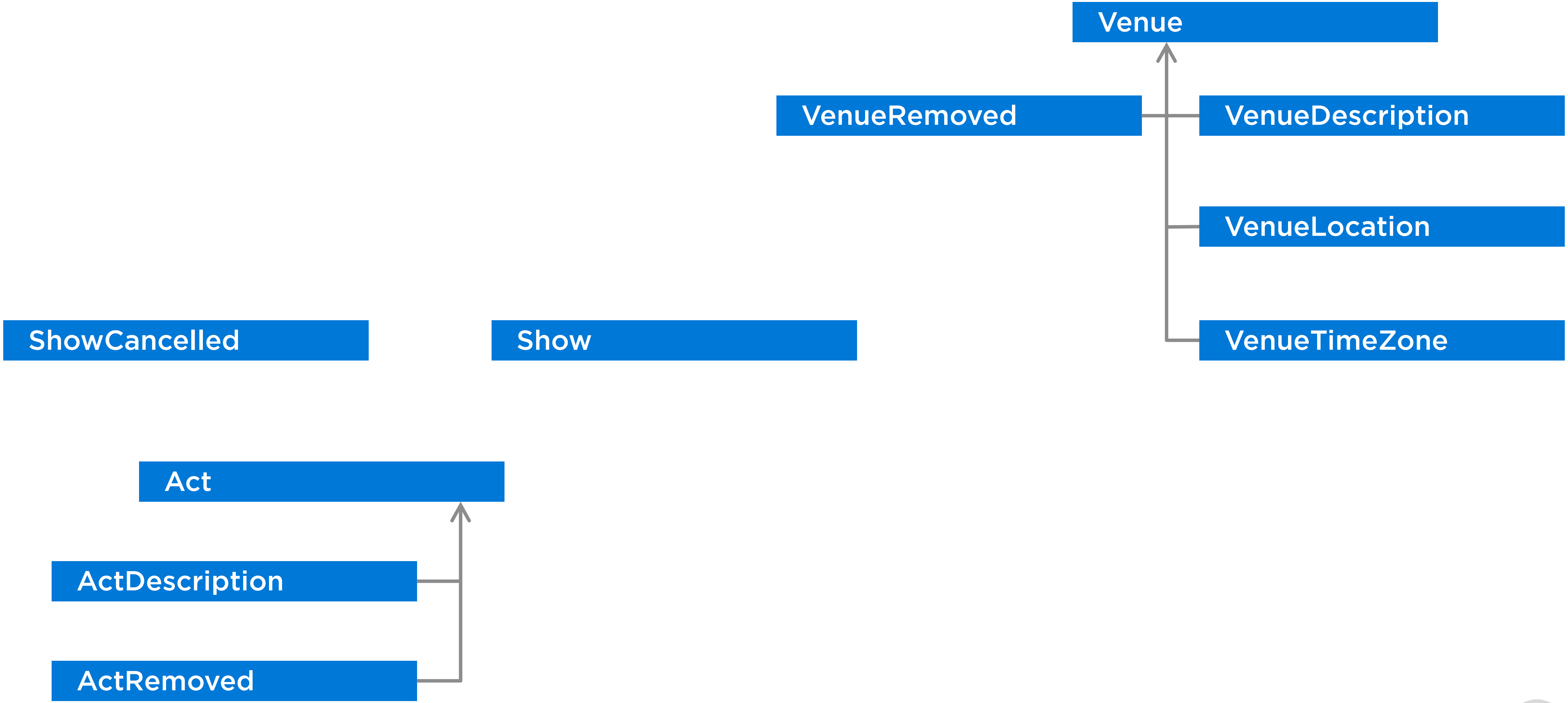
Show

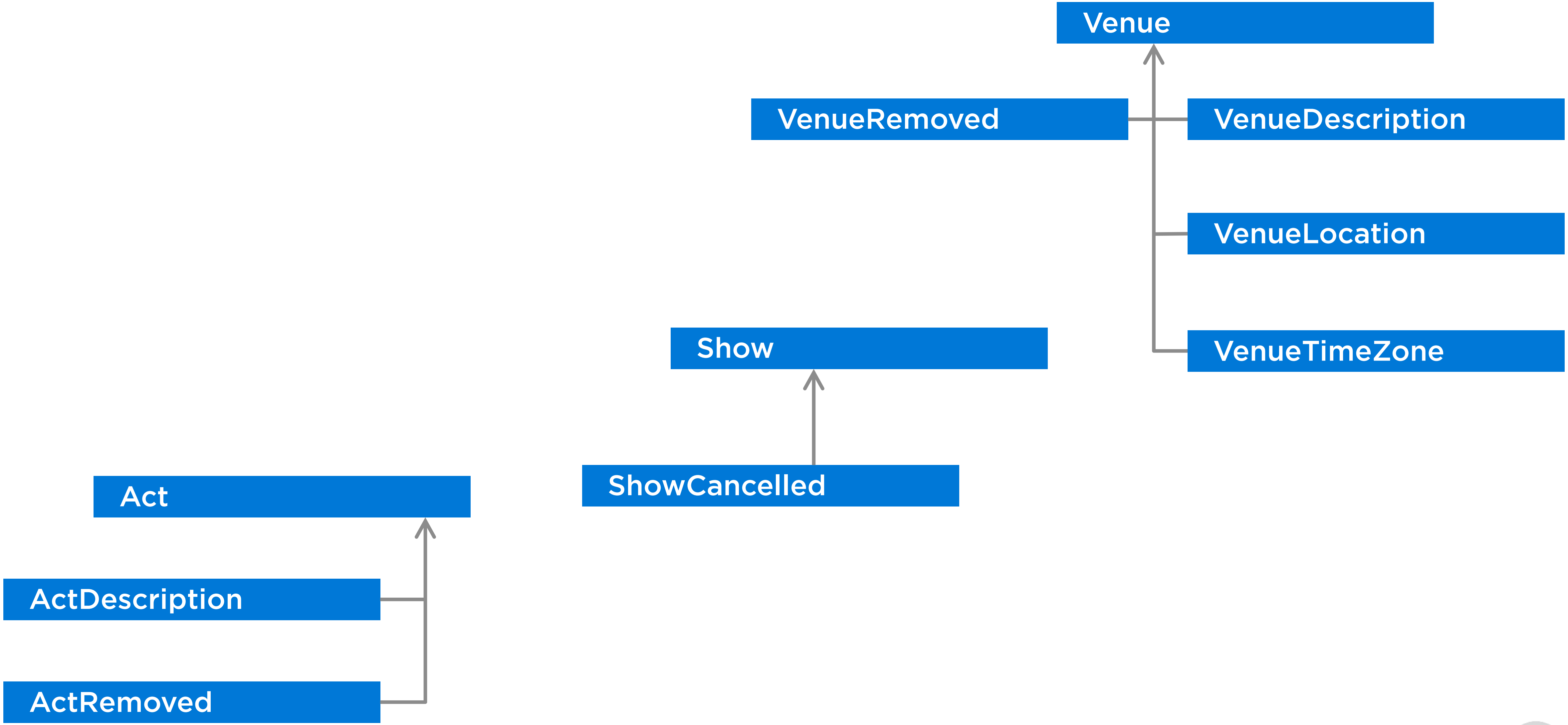
VenueTimeZone

Act

ActRemoved

ActDescription


Venue
VenueRemoved
VenueDescription
VenueLocation
ShowCancelled
Show
VenueTimeZone
Act
ActDescription
ActRemoved



Venue

VenueRemoved

VenueDescription

VenueLocation

VenueTimeZone

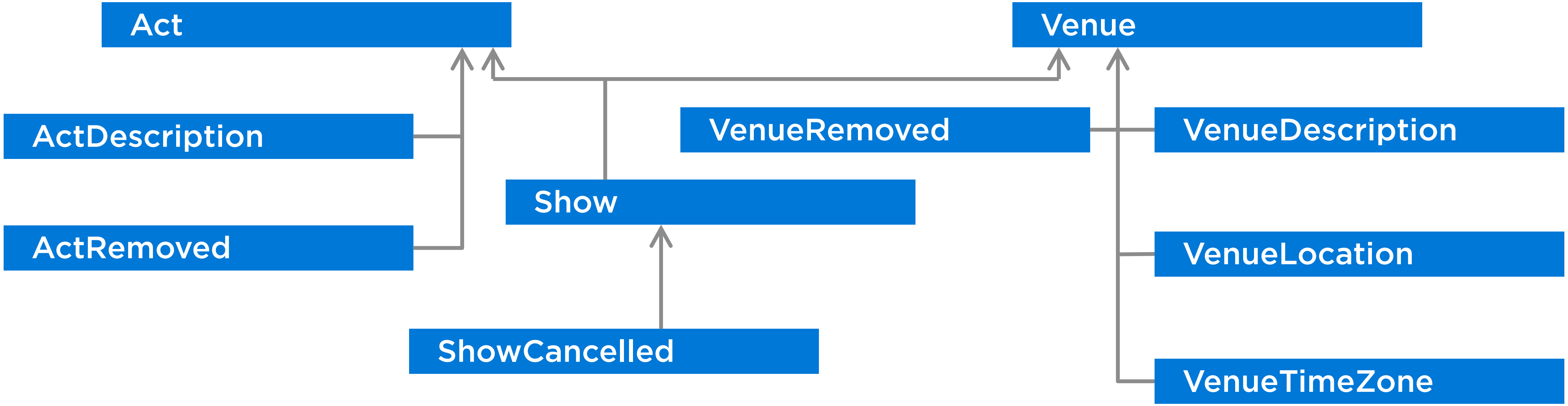
Show

ShowCancelled

Act

ActDescription

ActRemoved



Act

Venue

ActDescription

VenueRemoved

VenueDescription

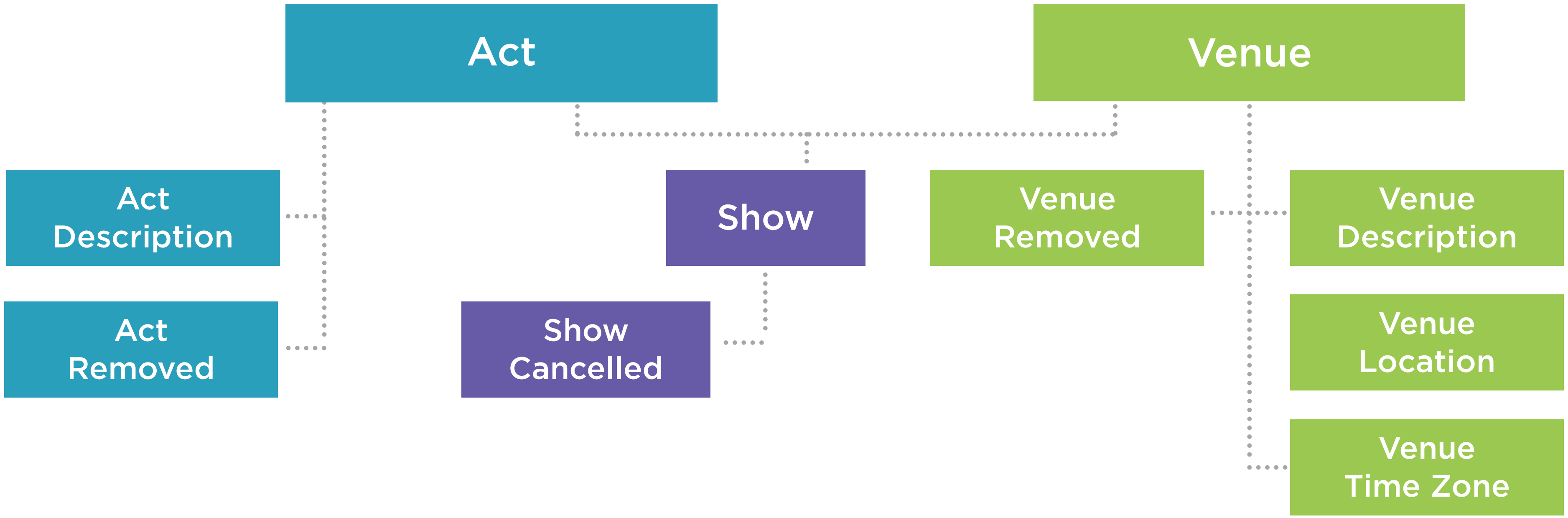
Show

ActRemoved

VenueLocation

ShowCancelled

VenueTimeZone


Act

Venue

Act Description

Act Removed

Show

Show Cancelled

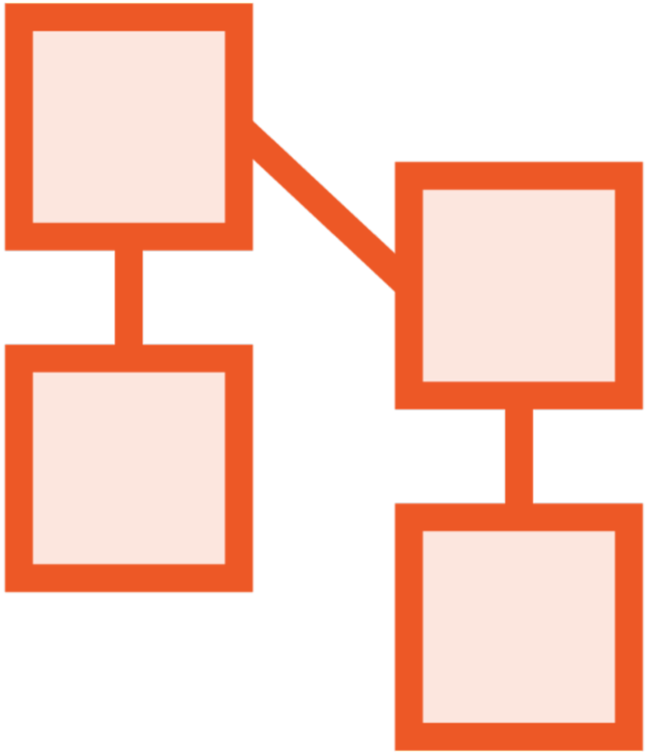
Venue Removed

Venue Description

Venue Location

Venue Time Zone

Keep Diagrams Small



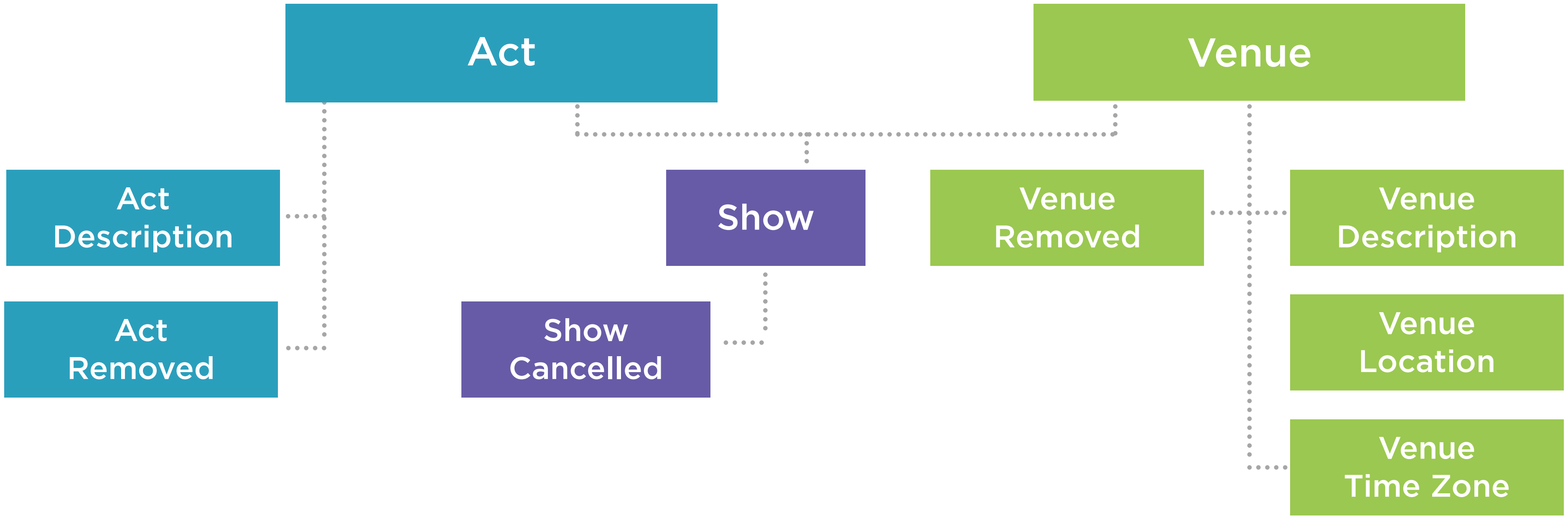
Contextual

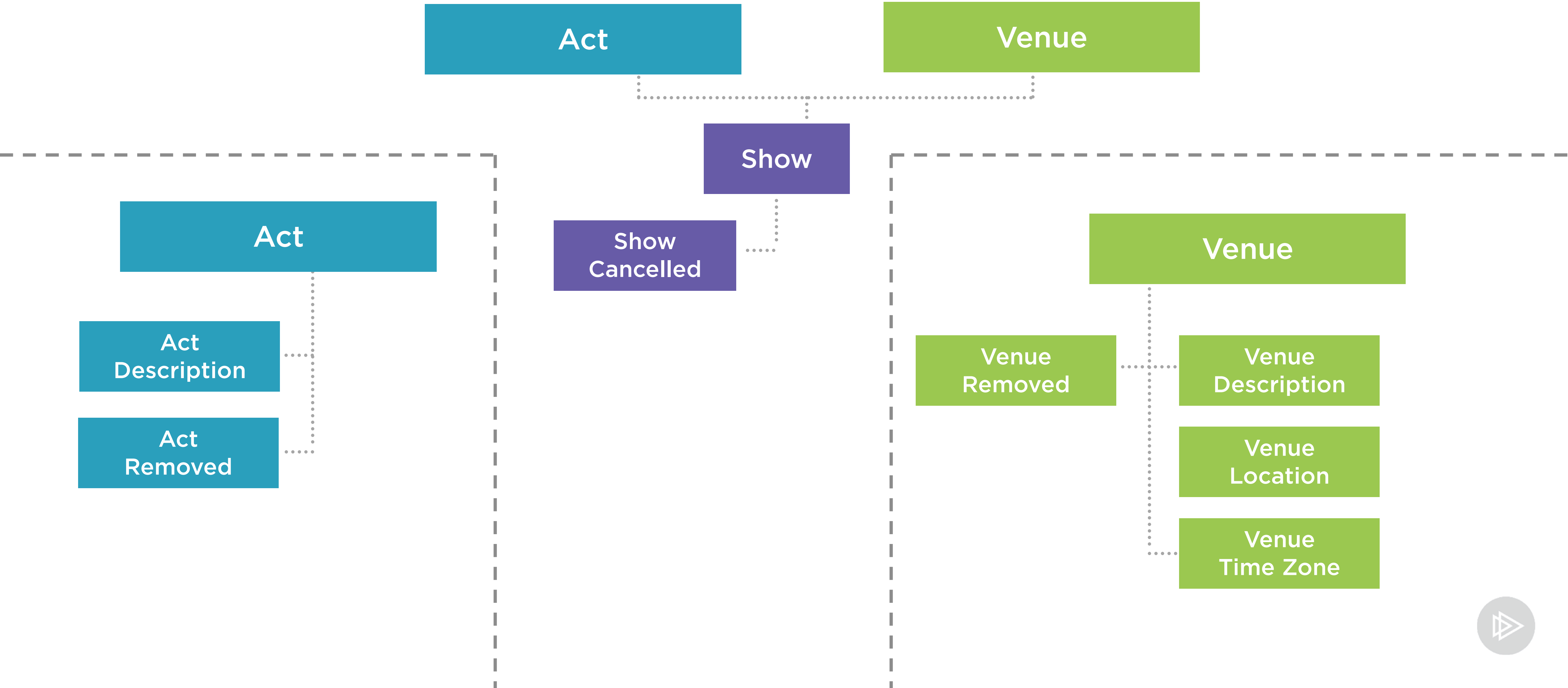
Microservice

Topic area

Near-top-level tables

Repeat parent tables

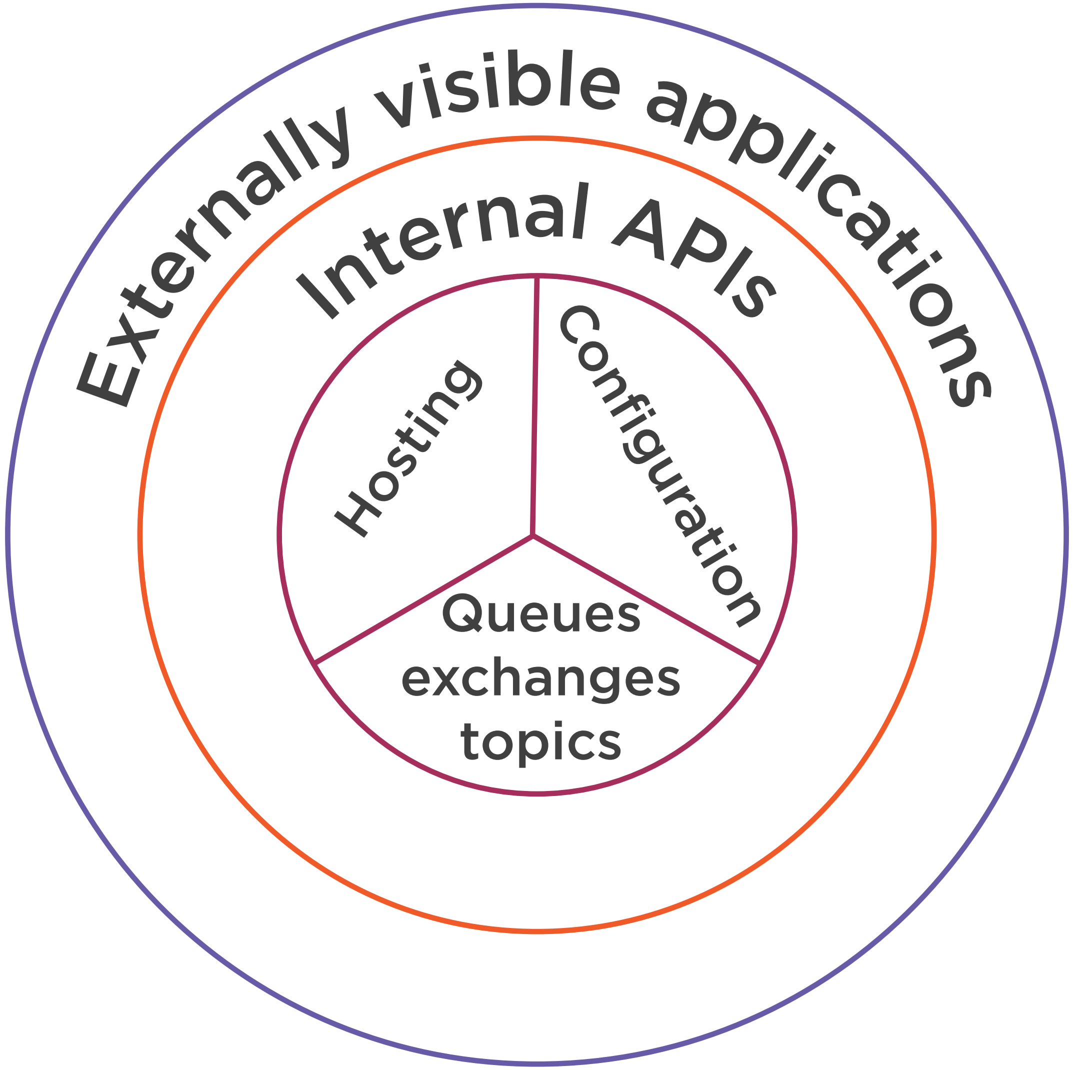

Act
Venue
Act Description
Show
Venue Removed
Venue Description
Act Removed
Show Cancelled
Venue Location
Venue Time Zone


Act
Venue
Show
Show Cancelled
Act
Act Description
Act Removed
Venue
Venue Removed
Venue Description
Venue Location
Venue Time Zone

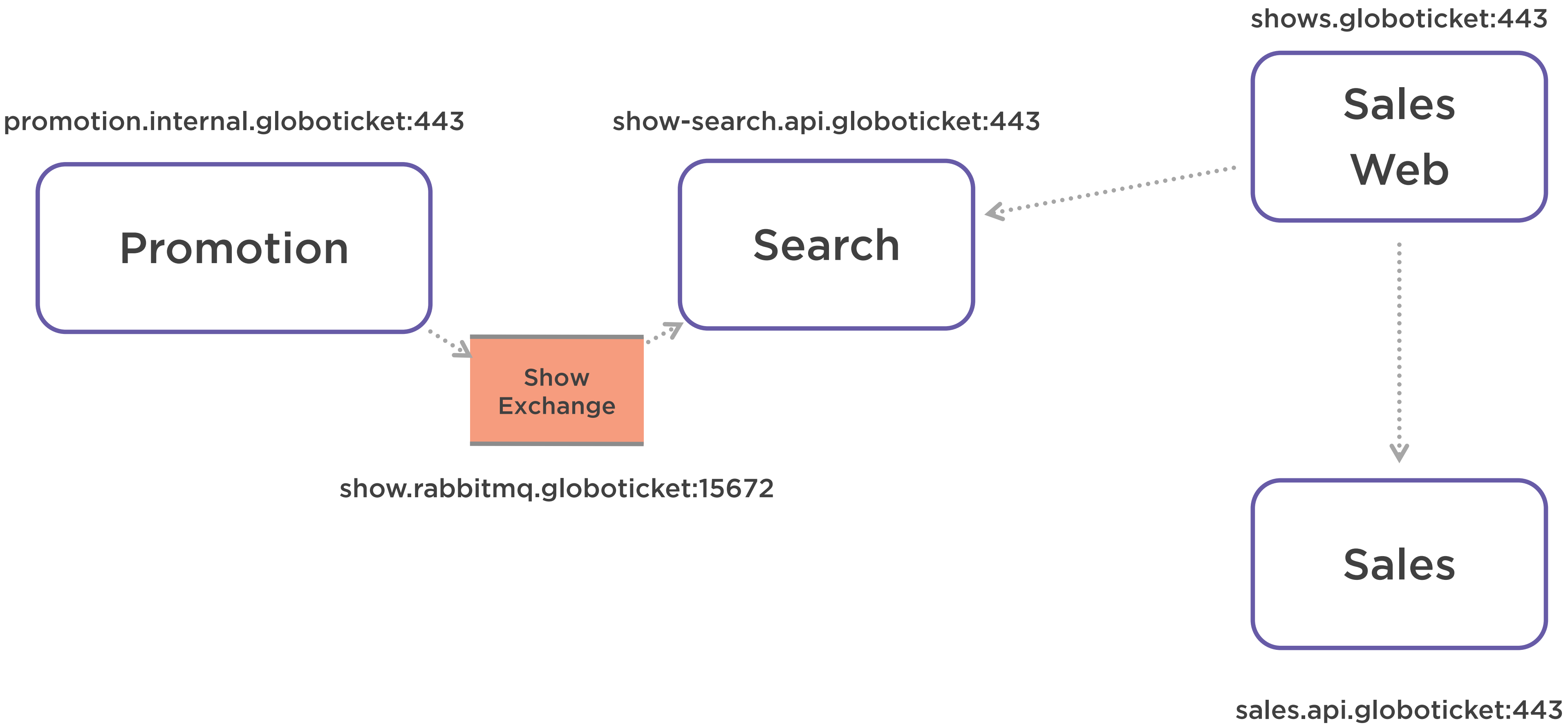
Application Dependency Map

- APIs
- Messages
- Machine names
- Configuration settings

Application Inventory



GloboTicket Application Dependency


shows.globoticket:443
Sales Web
promotion.internal.globoticket:443
show-search.api.globoticket:443
Promotion
Search
Show Exchange
show.rabbitmq.globoticket:15672
Sales
sales.api.globoticket:443

Inventory Backlog

Discover new applications

Review periodically

Coordinate reviews with releases

Problems and Mitigations

Problem	Mitigation
Endpoint /acts response time exceeds 250ms	In database promotion.sql, exec sp_updatestats
Queue show.rabbitmq Indexer exceeds 20 messages	Ensure that elasticsearch.shows is responding
Failed messages in show.rabbitmq Emailer_Error	Ensure that smtp.customer_service is responding

Development Practices

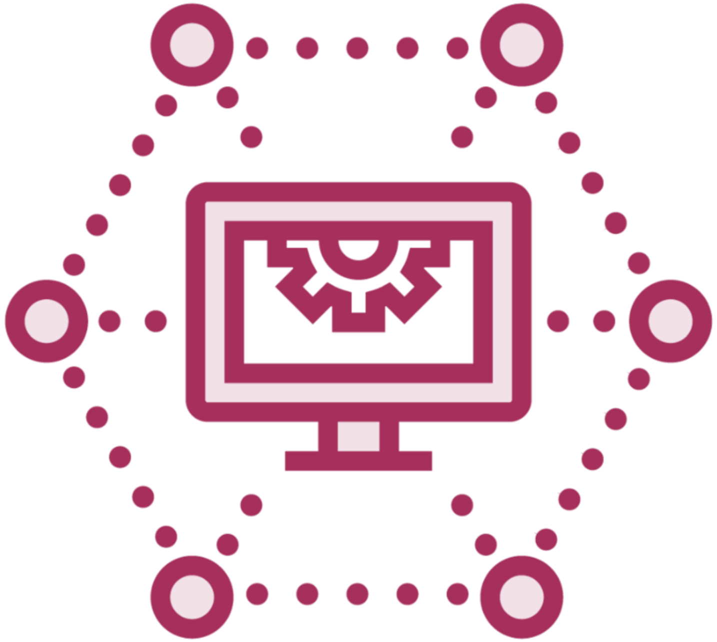
Happy path

Use runbook in QA

Ideally, fix problems immediately

Alternatively, document and log as technical debt

Production Practice

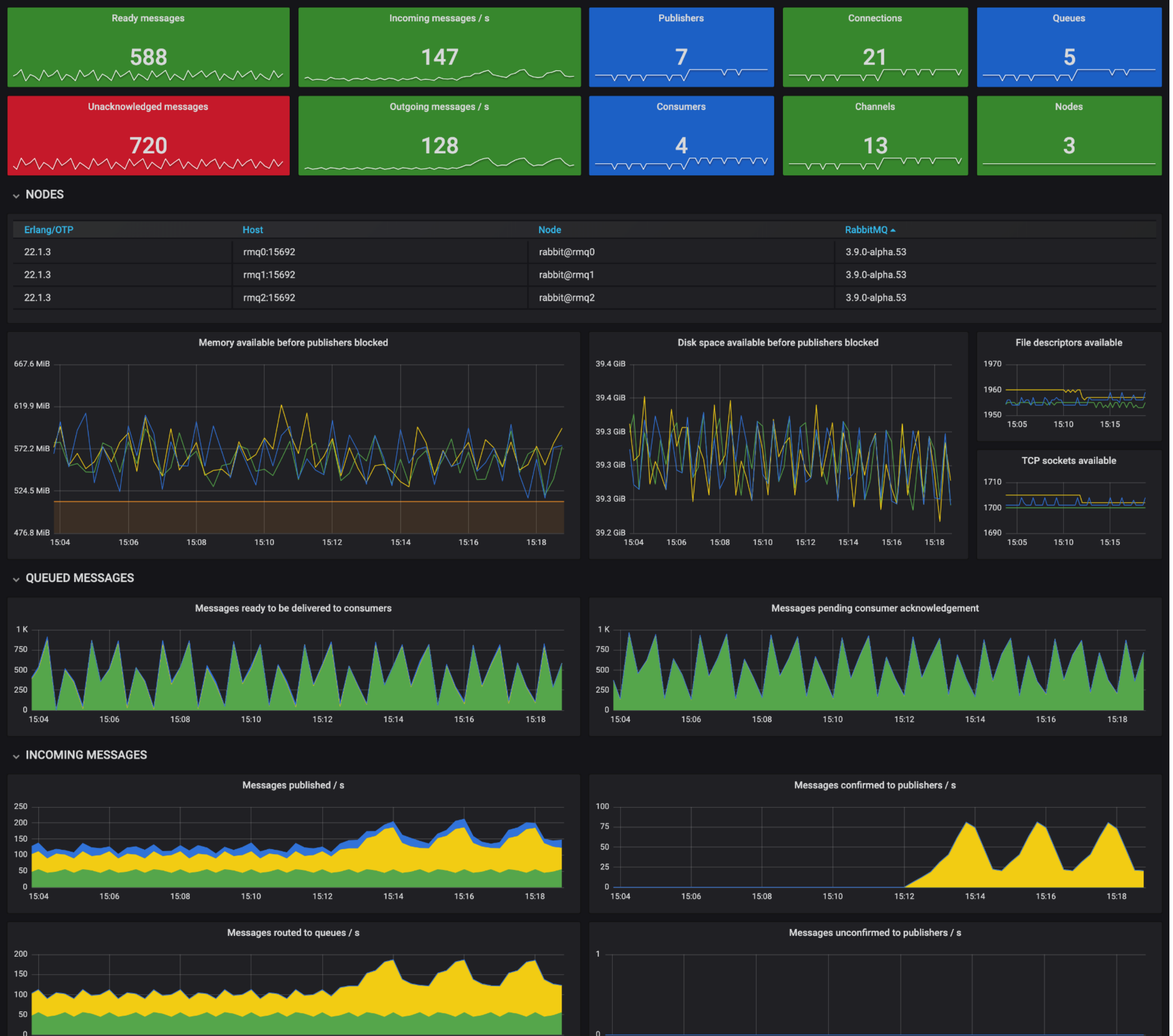


Cross-functional research team

Document diagnostic and corrective steps

Review material

Log problem/mitigation entries



Erlang/OTP	Host	Node	RabbitMQ ▲
22.1.3	rmq0:15692	rabbit@rmq0	3.9.0-alpha.53
22.1.3	rmq1:15692	rabbit@rmq1	3.9.0-alpha.53
22.1.3	rmq2:15692	rabbit@rmq2	3.9.0-alpha.53

Grafana dashboard showing RabbitMQ cluster metrics

https://grafana.com/grafana/dashboards/10991

RabbitMQ Metrics

Move messages from one application to another

Measure rate of messages into and out of queues

Message Delivery

Ready

Unacknowledged

Delivered

Node Health

CPU

Memory

Storage

Sockets

Cluster Usage

Connections

Channels

Queues

Consumers

Kafka Metrics

Replicating streams of messages

Partitions

In sync

Total

Performance

Request rate

Response rate

Delay

Request latency

Consumer lag

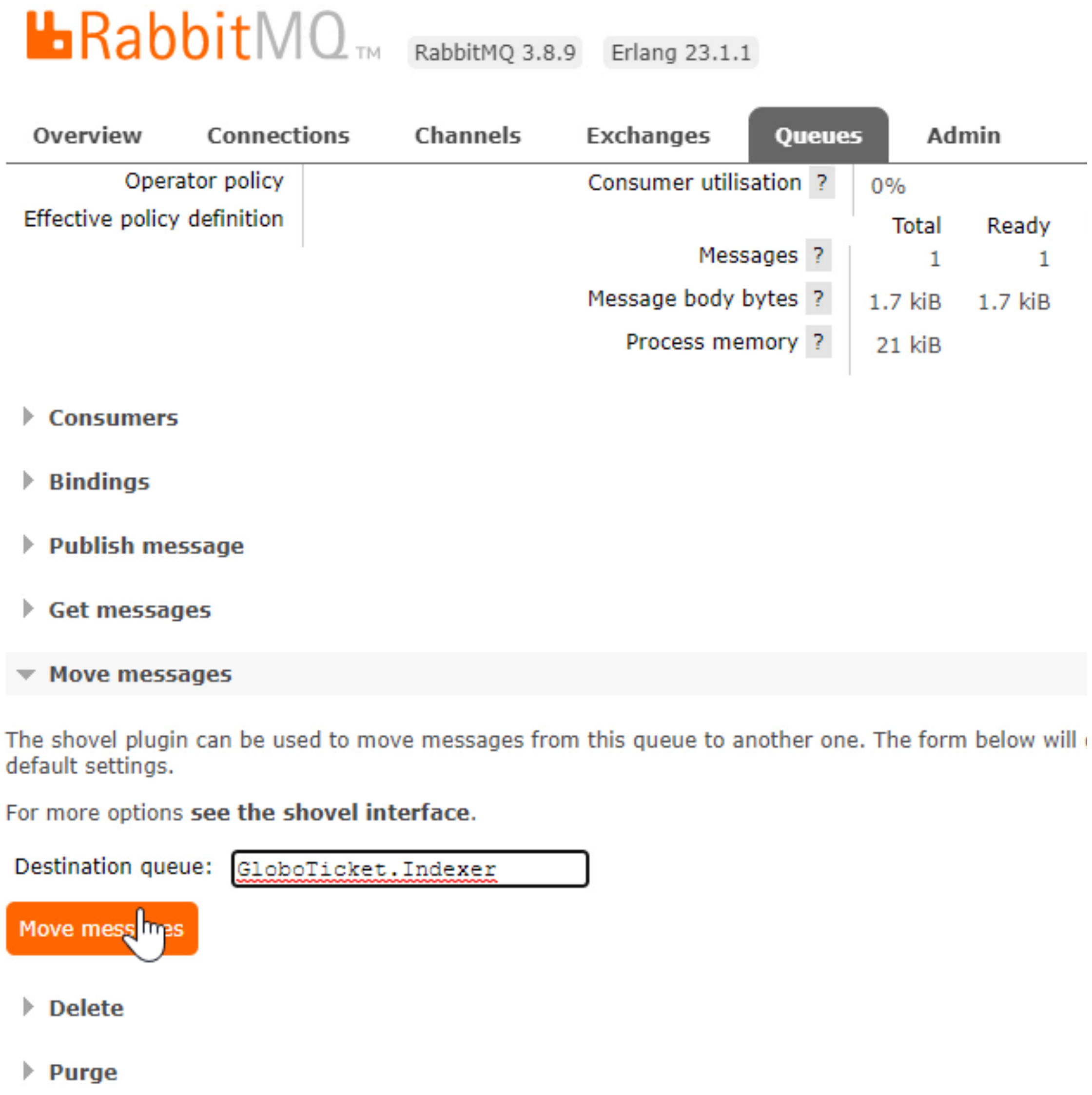
Monitoring and Dashboard Tools

Prometheus

Grafana

kibana

And many others based on platform



Operator policy

Effective policy definition

Consumer utilisation ? 0%

	Total	Ready
Messages ?	1	1
Message body bytes ?	1.7 kiB	1.7 kiB
Process memory ?	21 kiB	

The shovel plugin can be used to move messages from this queue to another one. The form below will
default settings.

For more options **see the shovel interface**.

Destination queue: GloboTicket.Indexer

Restoring messages from a dead letter queue

Failed Message Response

Alert

Locate

Diagnose

Correct

Restore

Monitor

Document

Effective Logging

Development

Context

Operations

No context

File Not Found or No Read Permission

Which is it?

Which file?

Where?

```
if (condition)
{
  try
  {
    var config = Read("ConfigSetting");
    DoThingThatMightFail(params, config);
  }
  catch (Exception ex)
  {
    throw new DetailedException(
      $@"Tried to do thing because
        condition with ConfigSetting:
        {params}, {config}. {ex.Message}“
    );
  }
}
```

◀ Why we tried

◀ Configuration setting

◀ What went wrong

◀ Wrap exception

◀ Attach parameters

Logging Standards

Verify in code reviews

Entry points

Inputs (before parsing)	info
Detail	debug
Return	debug

Exit points

Outputs	info
Detail	debug
Return	debug

Storage

SQL and parameters	debug

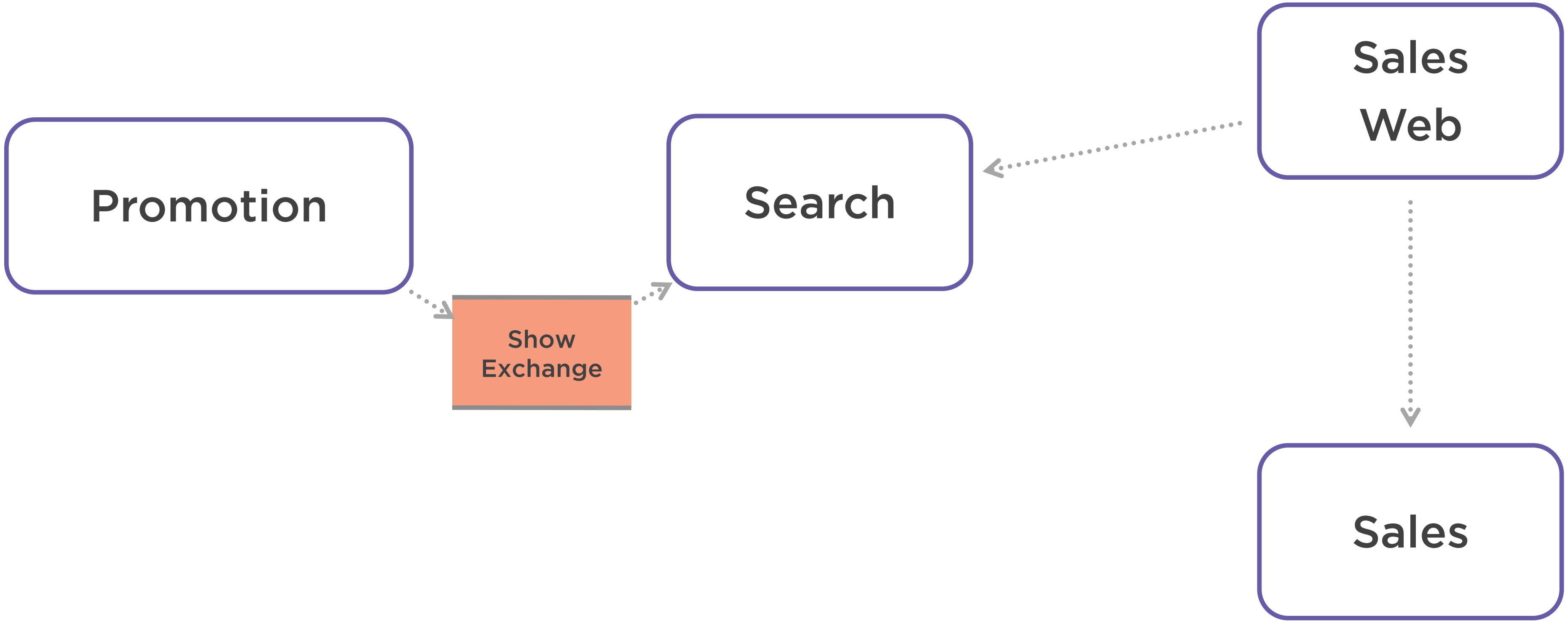
Decisions

Parameters	info

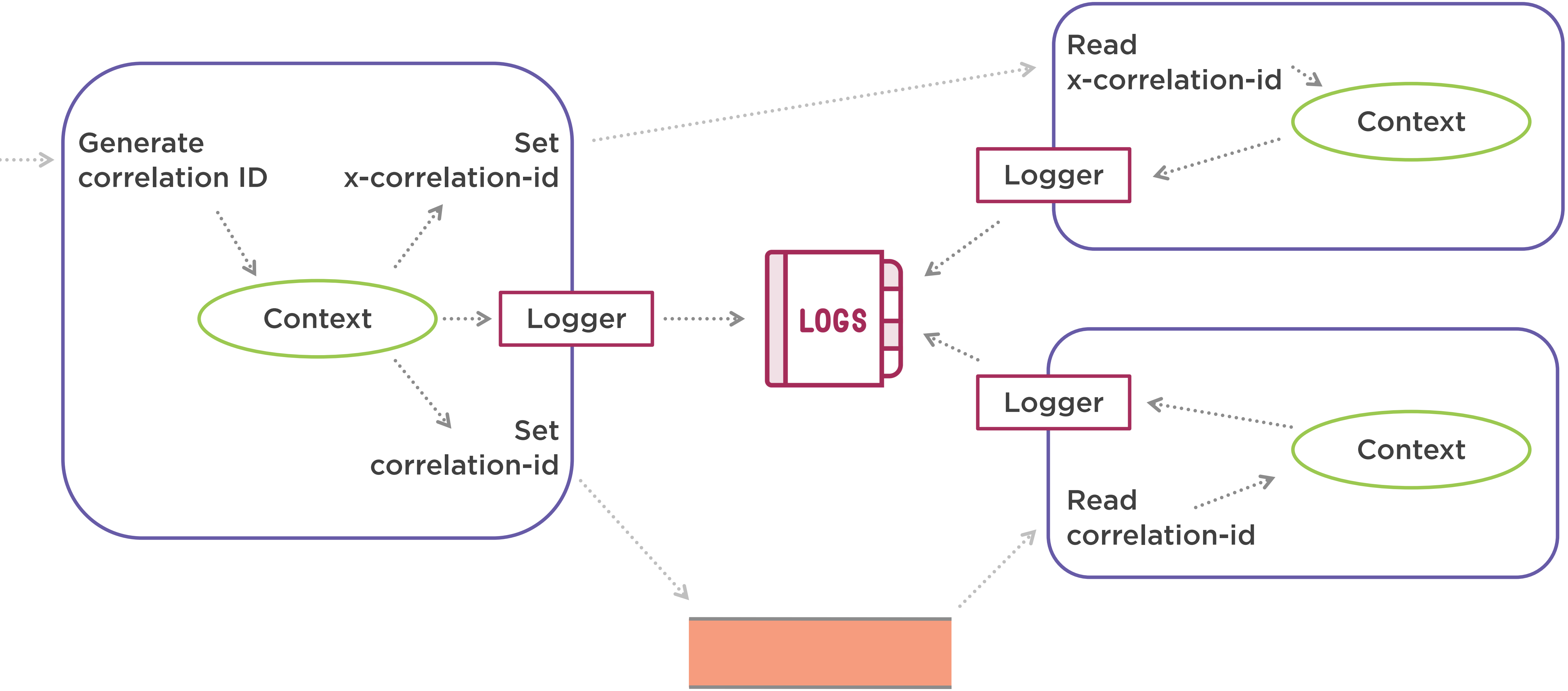
Exceptions

Context	error

Follow the Log



Correlated Tracing



Log Query

```
fields @log
| parse @log '"correlation_id":"*"' as correlation_id
| filter correlation_id = "123"
```

Deliberate Practices

Runbook

Developed with code
Tested in QA
Updated each release

Queues

Monitoring dashboard
Capture baseline
Dead letter queues

Logging

Standards and code reviews
Correlated tracing
Tested in QA

Up Next:
Managing Complex Scenarios