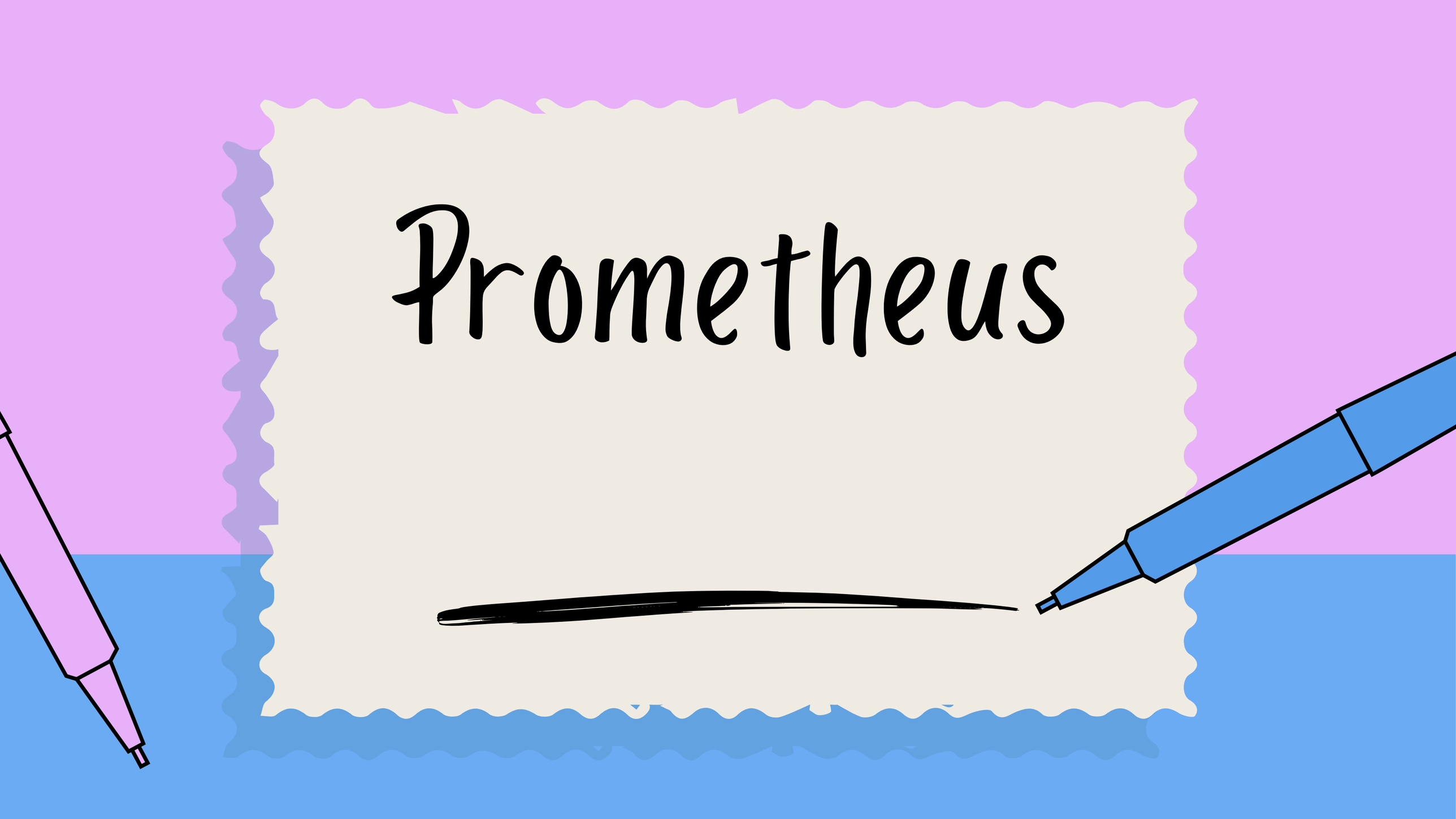
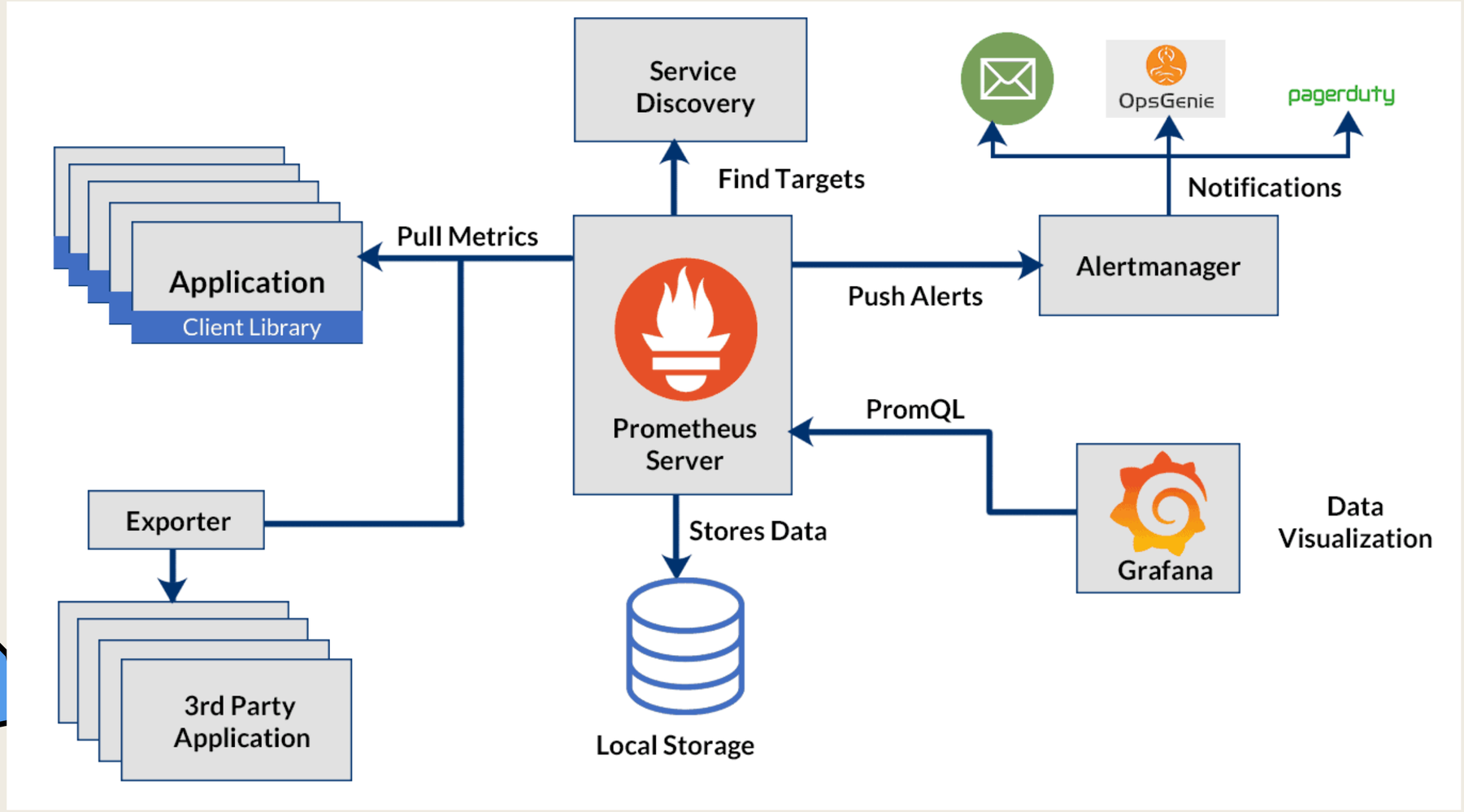
# Prometheus



# Architecture



# Metrics Collection Method

Prometheus exclusively uses a pull-based metric collection method. This means the Prometheus server actively fetches metrics from configured targets at regular intervals

How it works:

- Define targets
- Periodic scraping
- Metric exposure
- Data parsing and storing

# Metric Types

1. Counter(increment, reset)
2. Gauge(up, down)
3. Histogram(ranges in buckets)
4. Summary(quantiles)

# custom metric

any metric that is specifically defined and instrumented by you to monitor and observe specific aspects of your application or system that are not covered by default metrics.

Steps:

1. Choose metric type
2. Instrument your code
3. Expose the metrics endpoint
4. Configure Prometheus to scrape your application

# Best Practices

- Use base units in your metrics
- Name of metric starts with app name
- Suffix is the base unit(don't mix)
- Use labels for different request/stages
- Put the metric type
- Use tools for targets discovery
- Compress data to reduce storage

# Local Deployment

1. Create FastAPI app with Dockerfile and requirements
2. Create prometheus yaml and Dockerfile
3. Create docker-compose.yaml for both of them
4. Write the command docker-compose up