

파워 앱과 서비스로 홈 트레이닝 앱 로그 데이터 추적하기

서비스 한국 사용자 모임 2021년 5월



파워 앱과 서비스로 홈 트레이닝 앱 로그 데이터 추적하기

서비스 한국 사용자 모임 2021년 5월



화려한 조명이



나를
감싸줄게

Justin Yoo

시니어 클라우드 아드보캣

Microsoft

Developer Relations



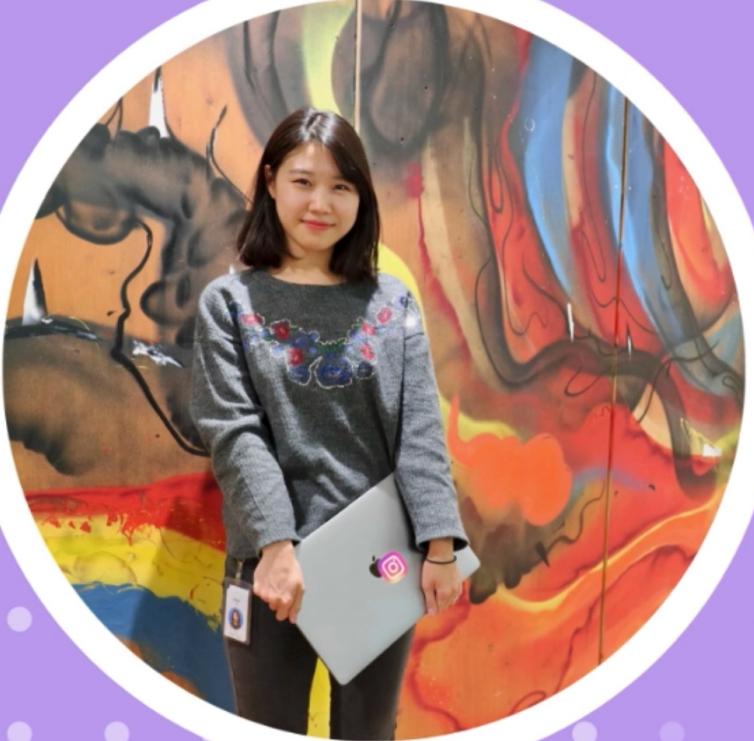
@justinchronicle



@aliencube.community



개발자가 서버리스로 운동하는 방법



서지연

오늘도 삽질하며 배우고 있는
백엔드 개발자

개발자가 SERVERLESS로 운동하는 방법

장기화되는 집콕과 운동 부족으로 급찐살, 이대로는 안되겠다 싶어서 운동은 시작했는데 여전히 힘들기만 합니다.

꾸준한 운동을 위한 동기부여!
없으면 내가 만들자!

트위터, OCR, 그리고 serverless 스택으로 만든
운동로그 자동화 프로젝트를 소개합니다.



닌텐도 린핏 어드벤처

앗! 링핏이 없는데요!



그래서 찾아봤습니다



Mallee Bulls Fitness Centre



안녕! 친구들~! 빽빽이 아저씨야!

운동 일지 기록용 파워 앱

Test Mode



Exercise

Full Body

Kettle Bell Swing

Sets

Reps

Weight

Weight

Unit kg lbs

Additional Note

Back

Go!

I'm Done!



New Item Update Discard Delete Upload Item



Items

SELECT * FROM c

Edit Filter

id /itemType

9c0c3de4-42a4-4b58-... Routine

d33ccaeef-e234-4193-... Routine

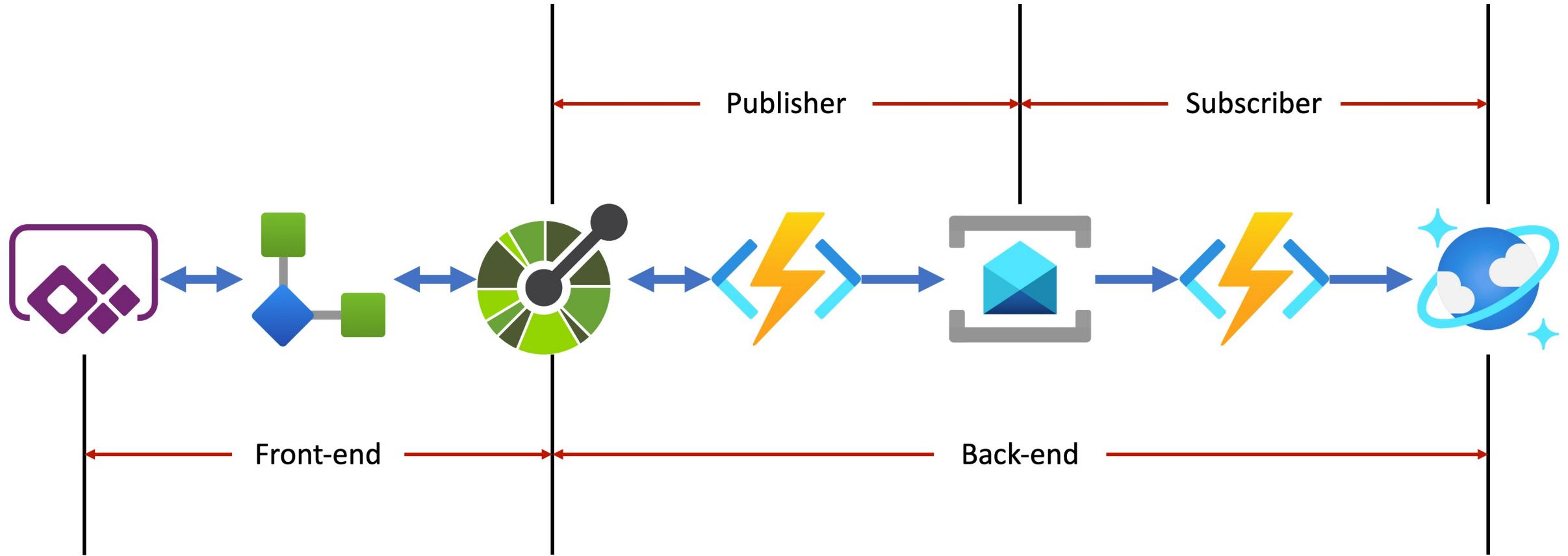
d10d74b6-536e-4913-... Routine

1b16df0d-4f68-48c3-... Routine

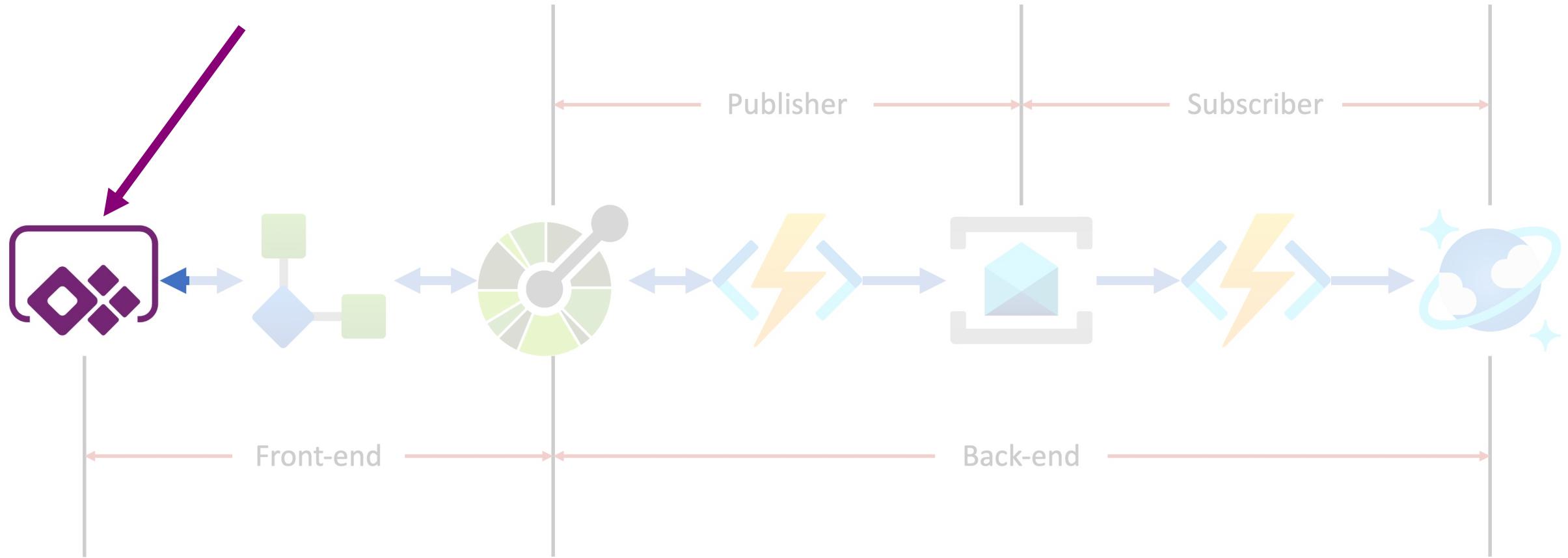
234e5f96-5a51-4a59-... Routine

Load more

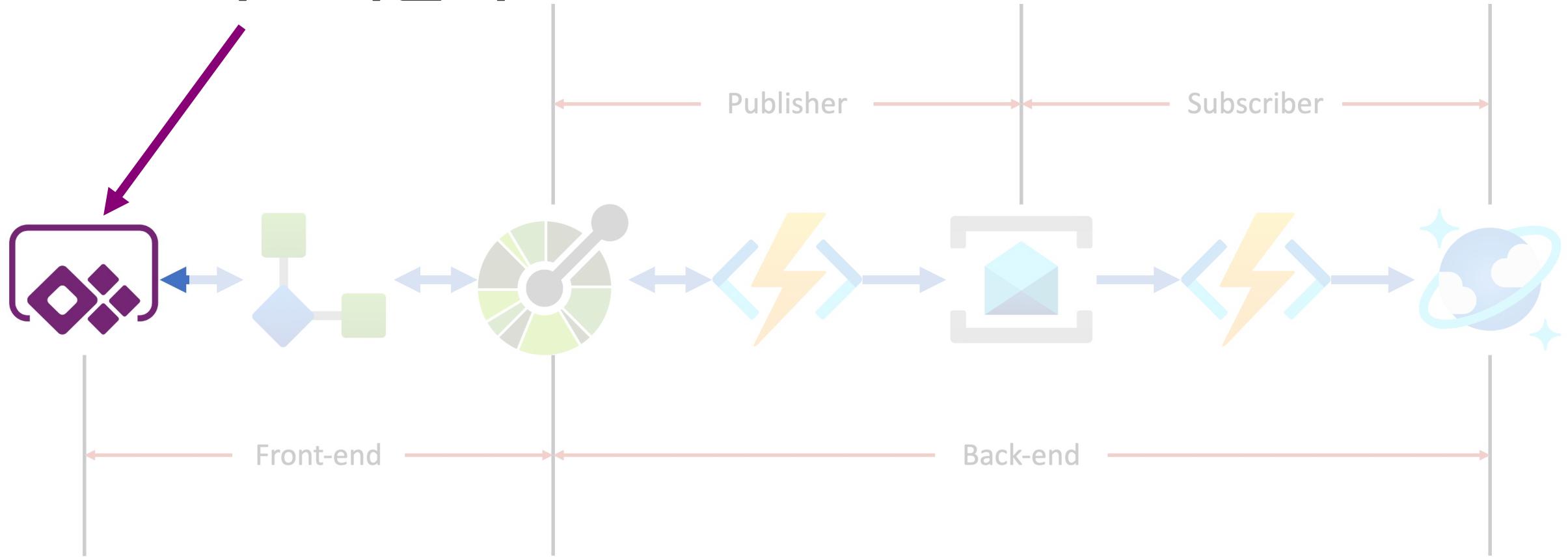
```
1  {
2      "upn": "...",
3      "correlationId": "94e5da6f-19ad-49e8-9673-34883108f95e",
4      "interface": "Test Harness",
5      "routineId": "e6b03975-e488-42b2-9d8a-806c459270bc",
6      "routine": "New Routine: 2021-05-10, 11:40",
7      "exercises": [
8          {
9              "exerciseId": "c85cd89d-1d60-4622-965c-0dd313b20349",
10             "name": "Dumbbell Lateral Raise",
11             "Target": "Shoulder",
12             "sets": [
13                 {
14                     "sequence": 1,
15                     "repetitions": 15,
16                     "weight": 4,
17                     "unit": "kg"
18                 },
19                 {
20                     "sequence": 2,
21                     "repetitions": 15,
22                     "weight": 4,
23                     "unit": "kg"
24                 }
25             ]
26         }
27     ]
28 }
```

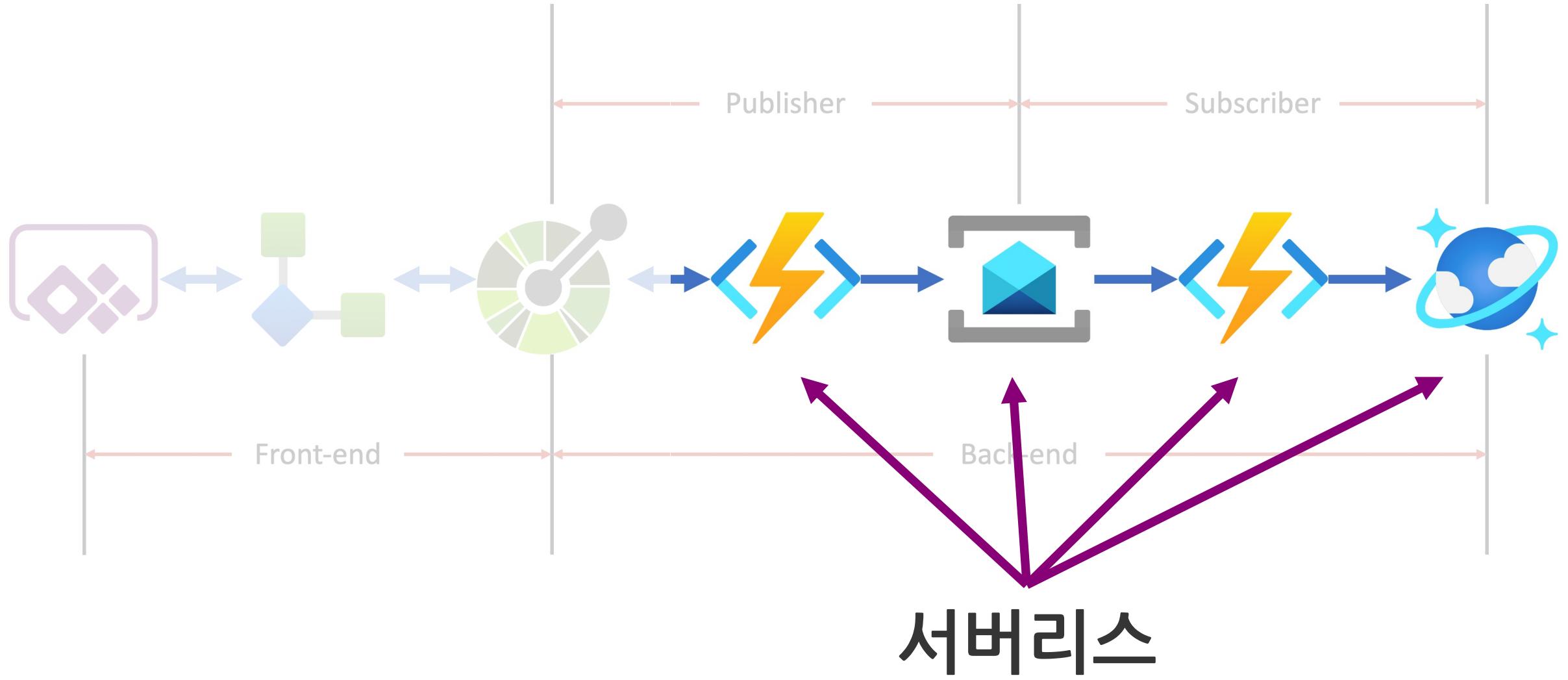


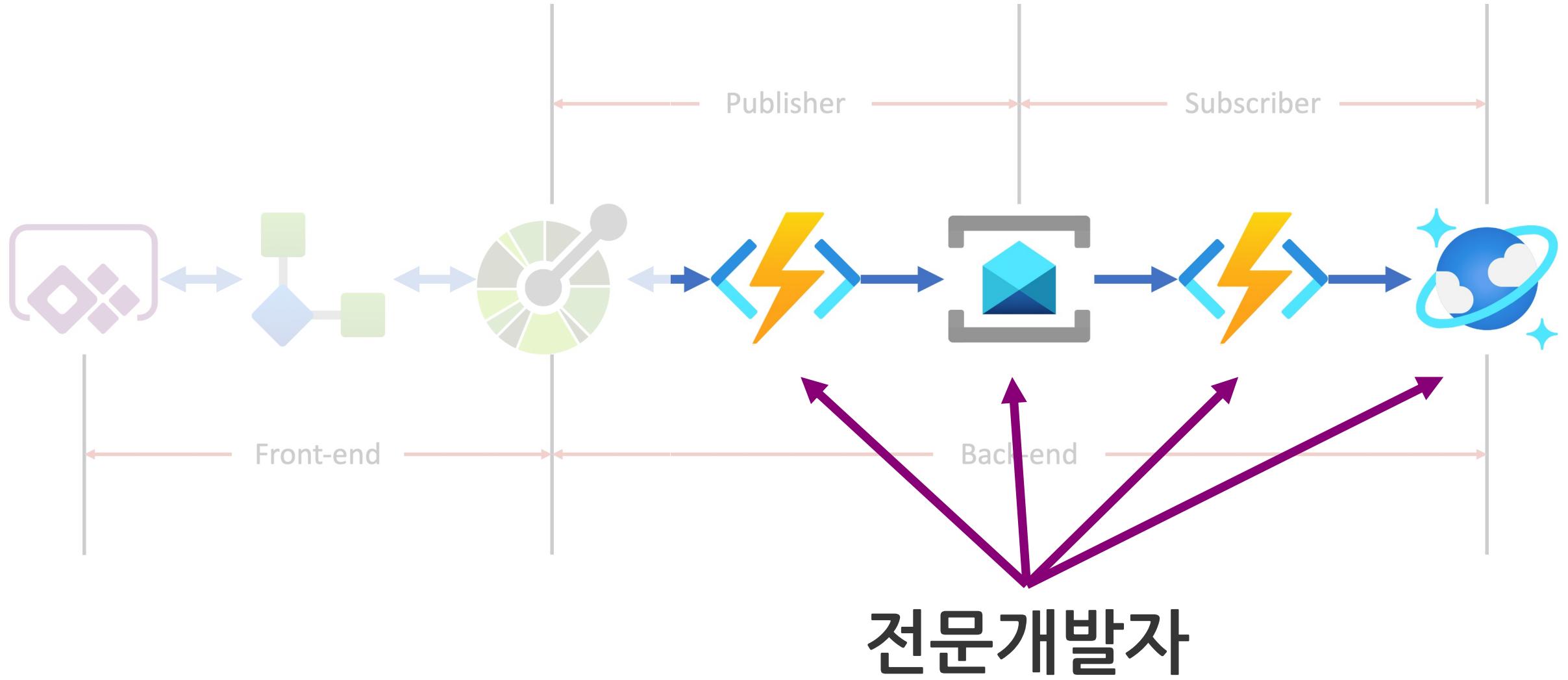
SaaS

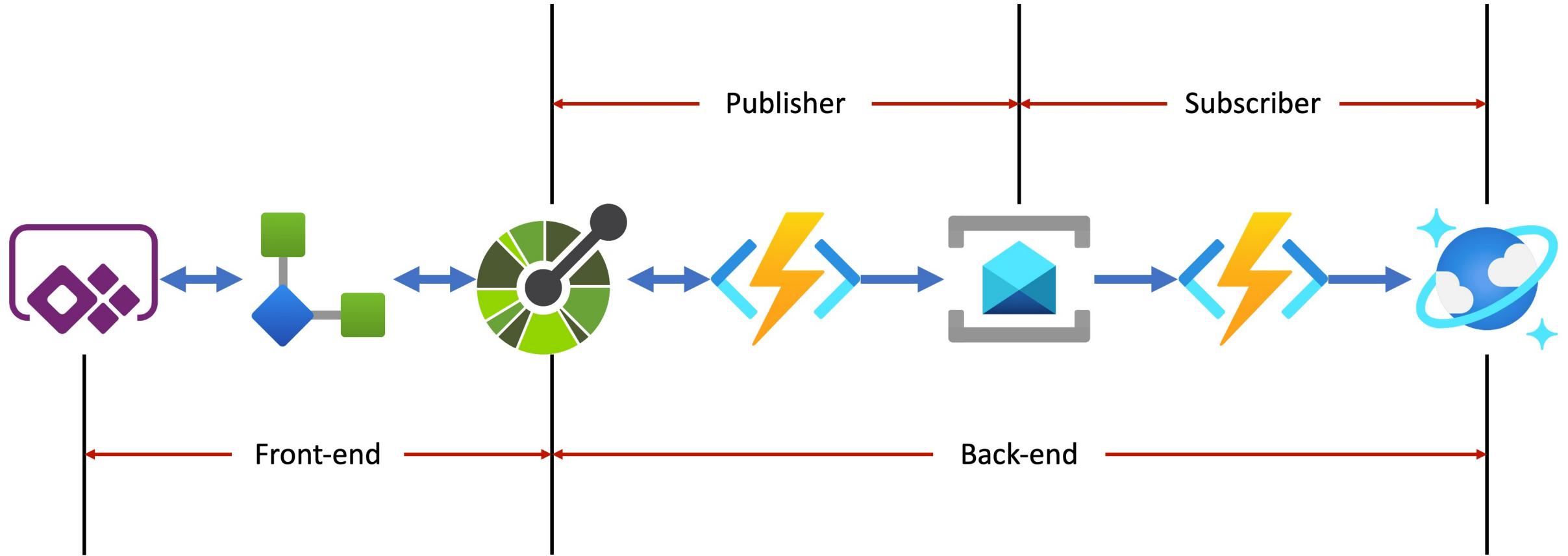


시민개발자









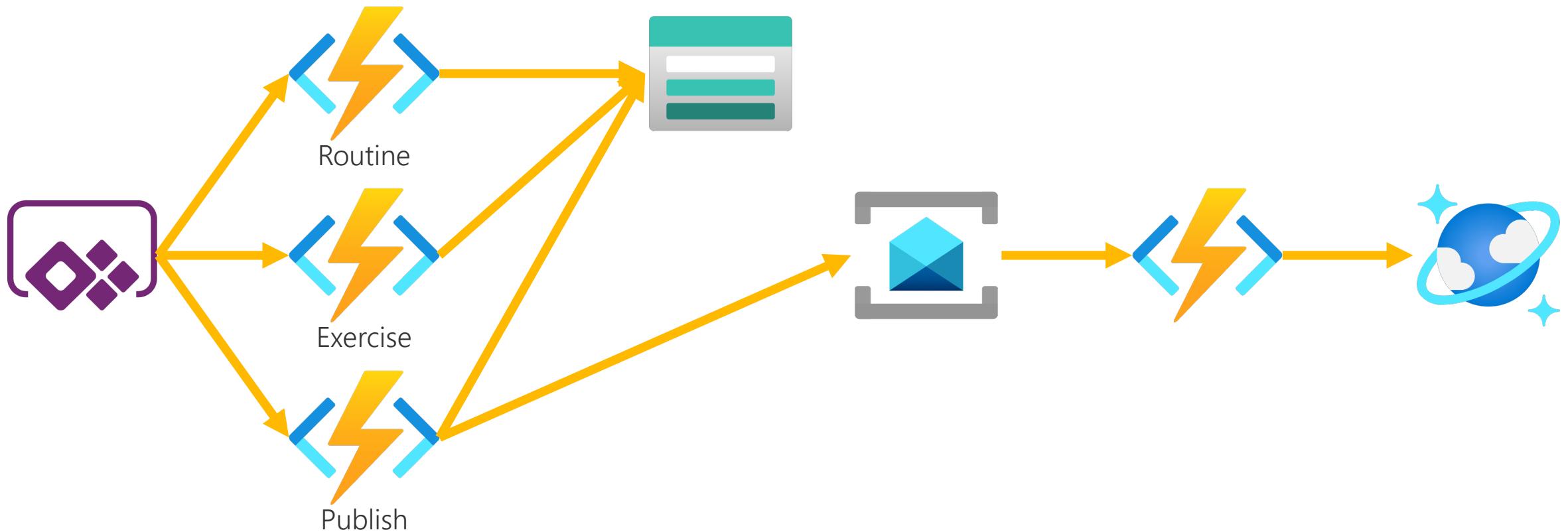
3대 500을 치는 그날을 위해!

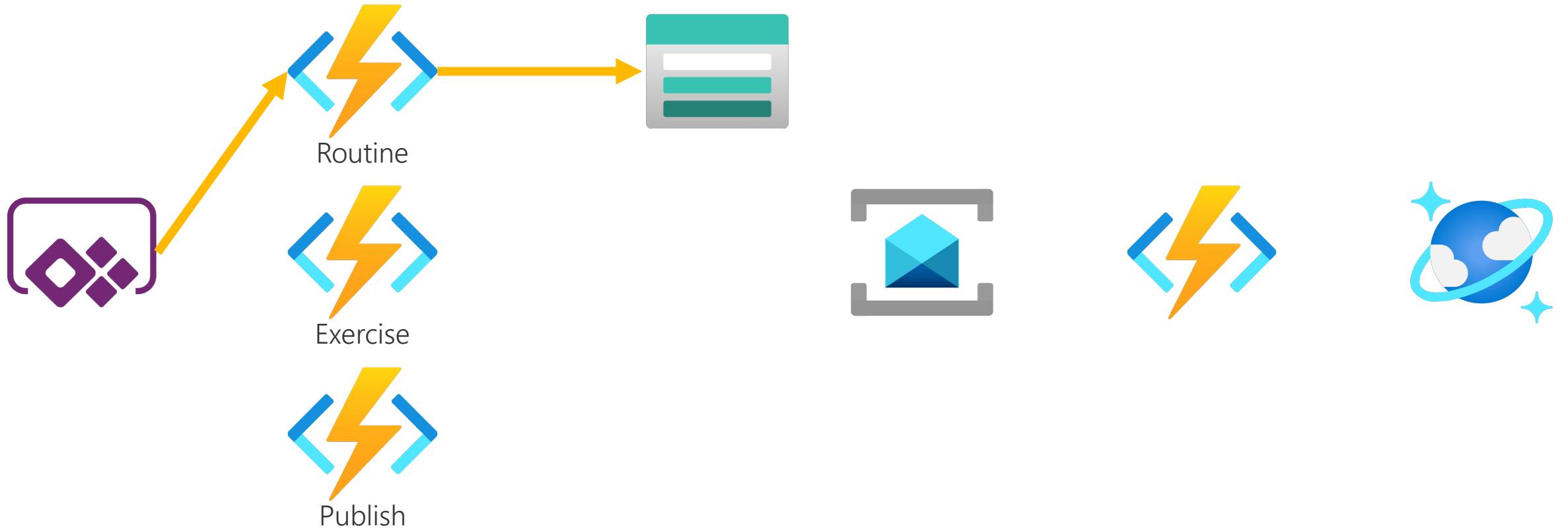
데모: 파워 앱 ➔ 코스모스 DB

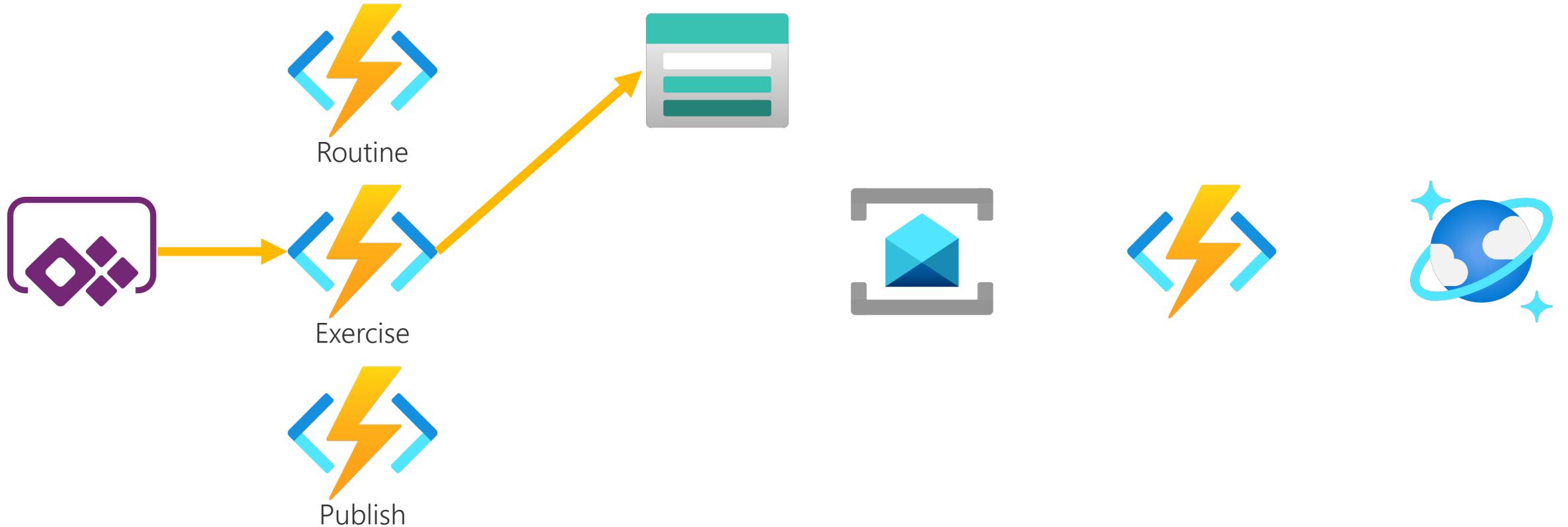
그런데 말입니다

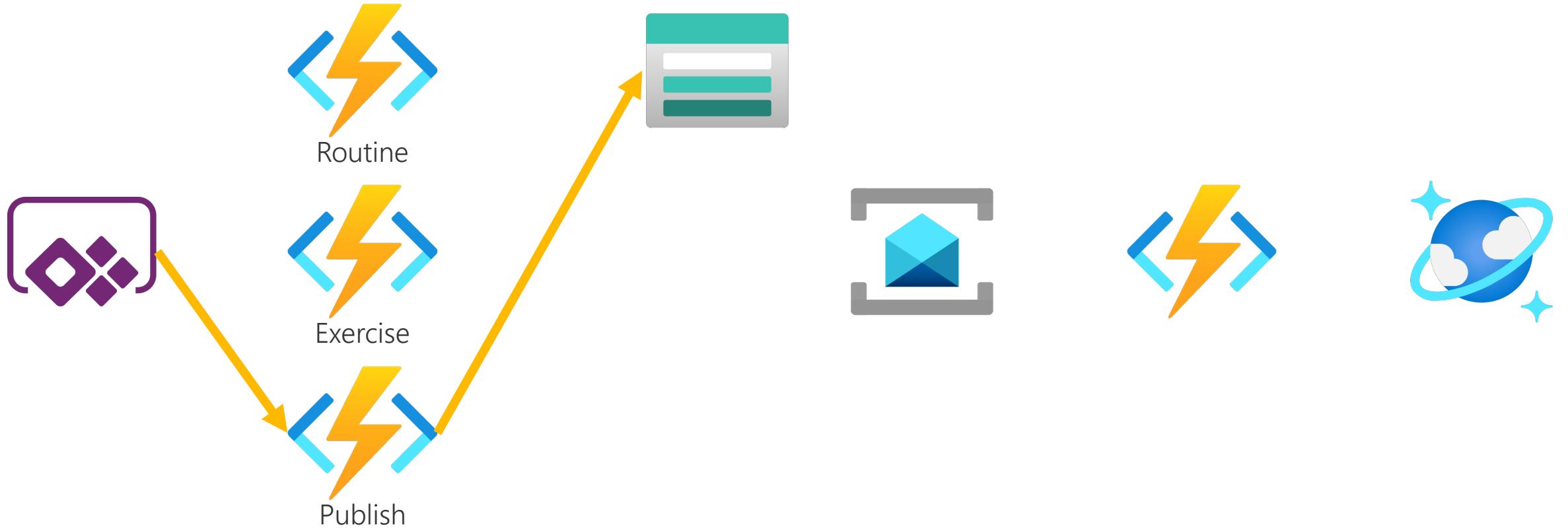
운동 일지 저장중 에러 발생!

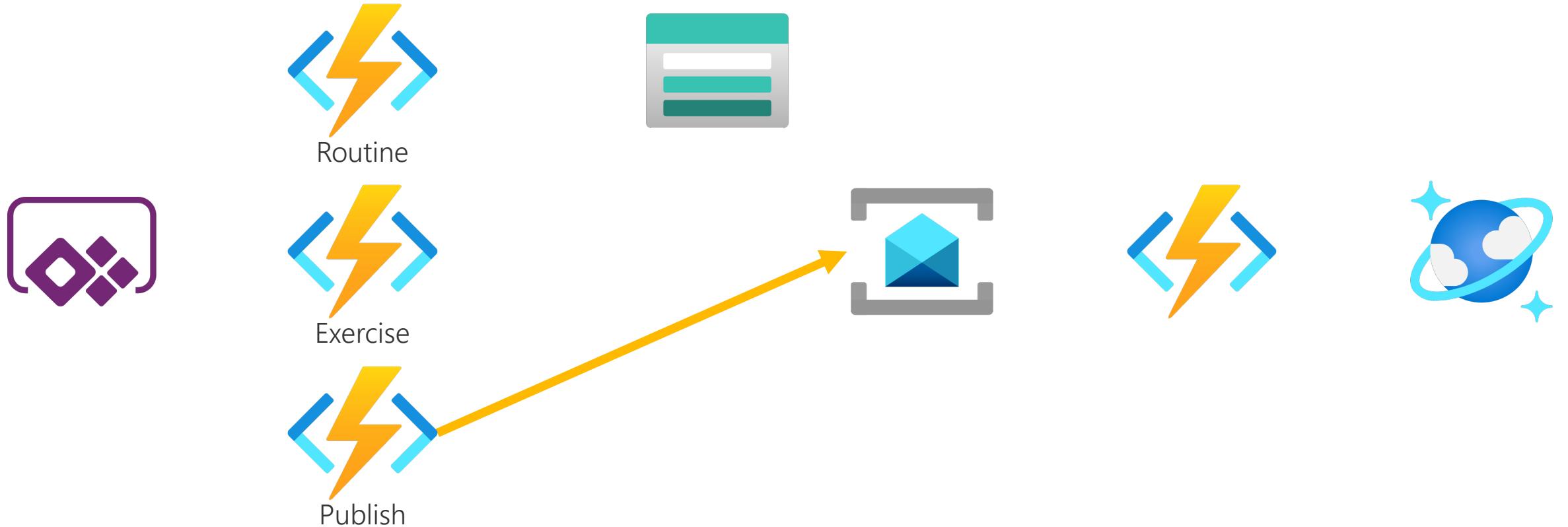
어디서 에러가 생기는지 모르겠음













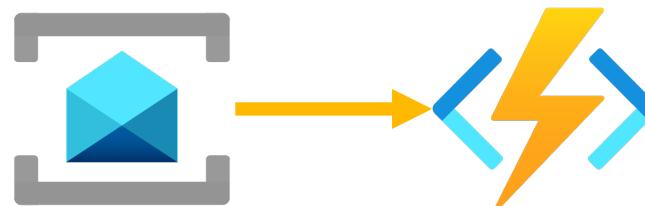
Routine



Exercise

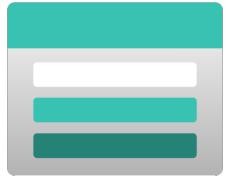


Publish





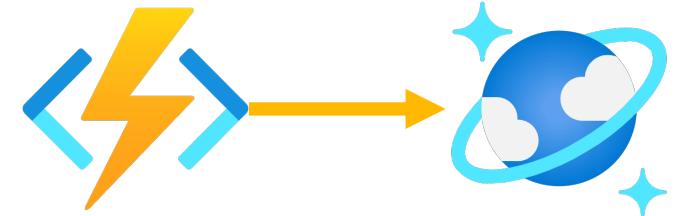
Routine



Exercise

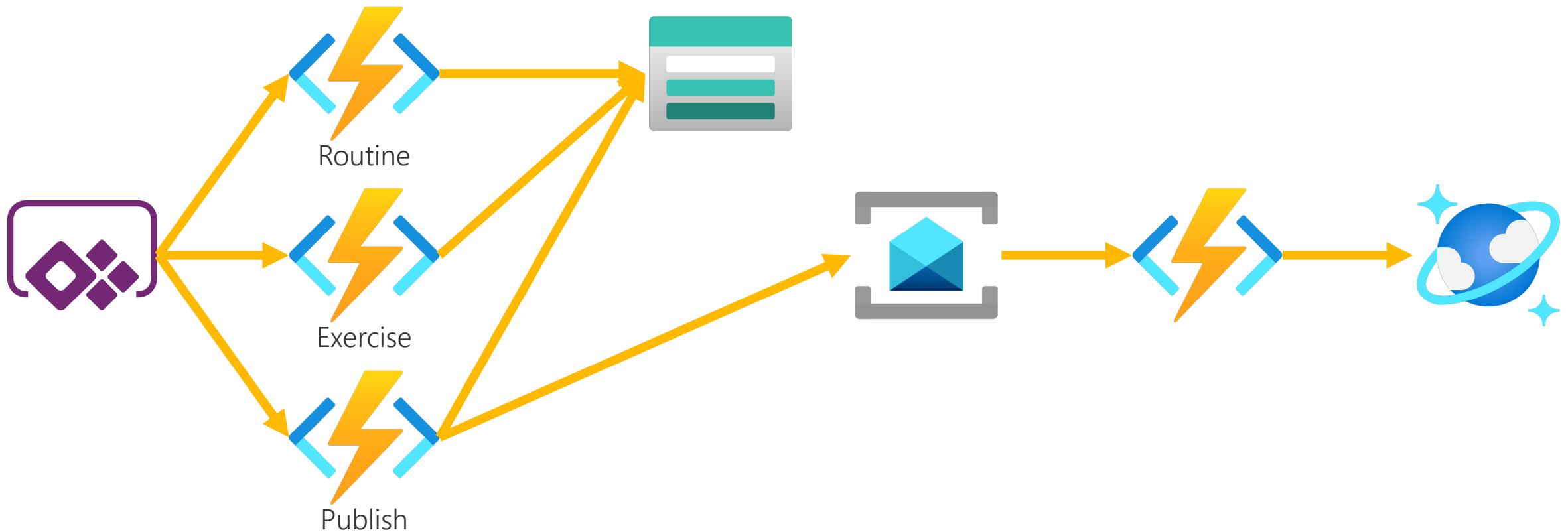


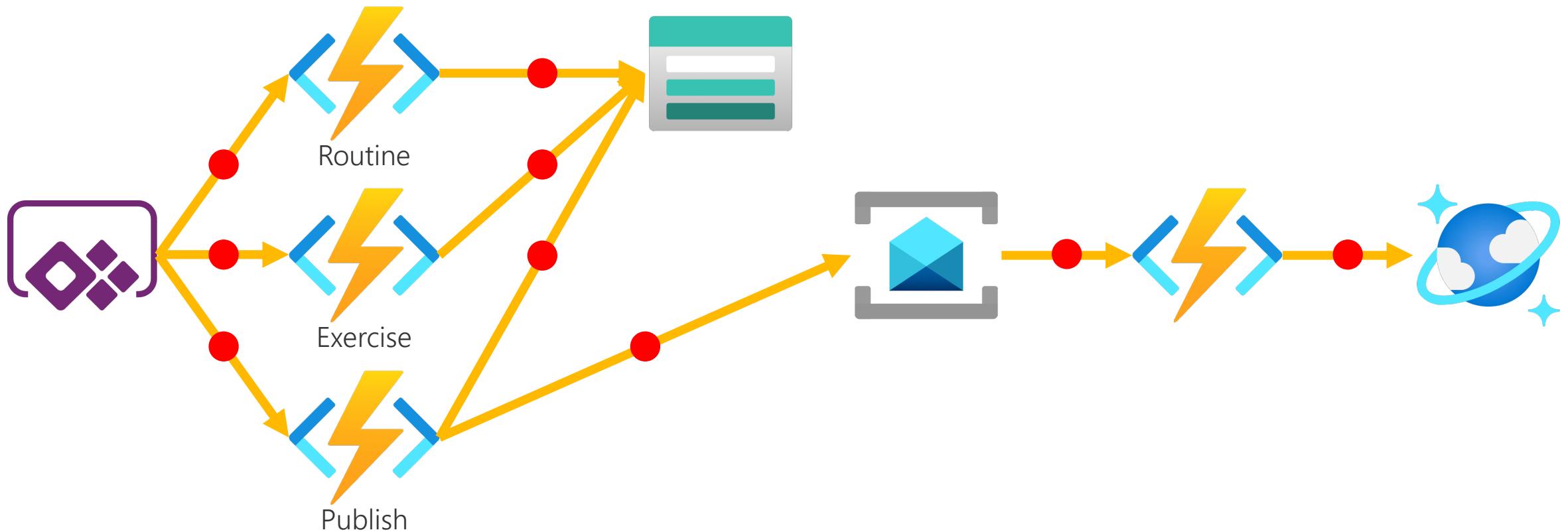
Publish

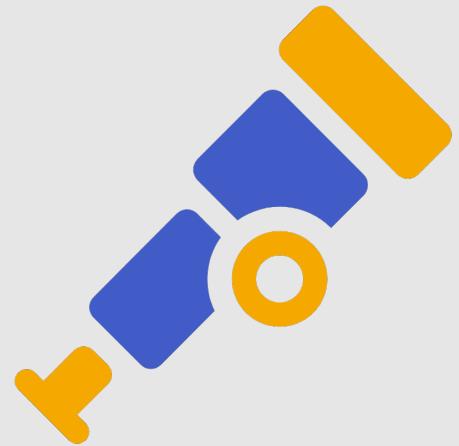


어디서 에러가 생기는지 모르겠음

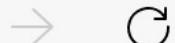
에러가 어디서 생기는지 체크포인트를 찍어보자







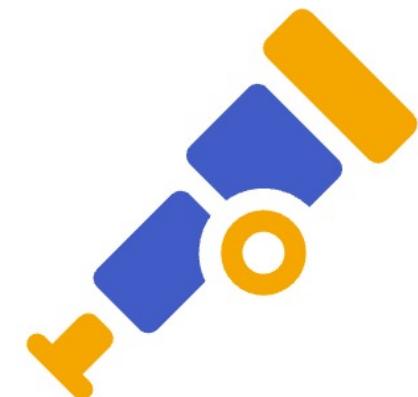
오픈 텔레메트리
<https://opentelemetry.io>



https://www.cncf.io/sandbox-projects/



...



OpenTelemetry

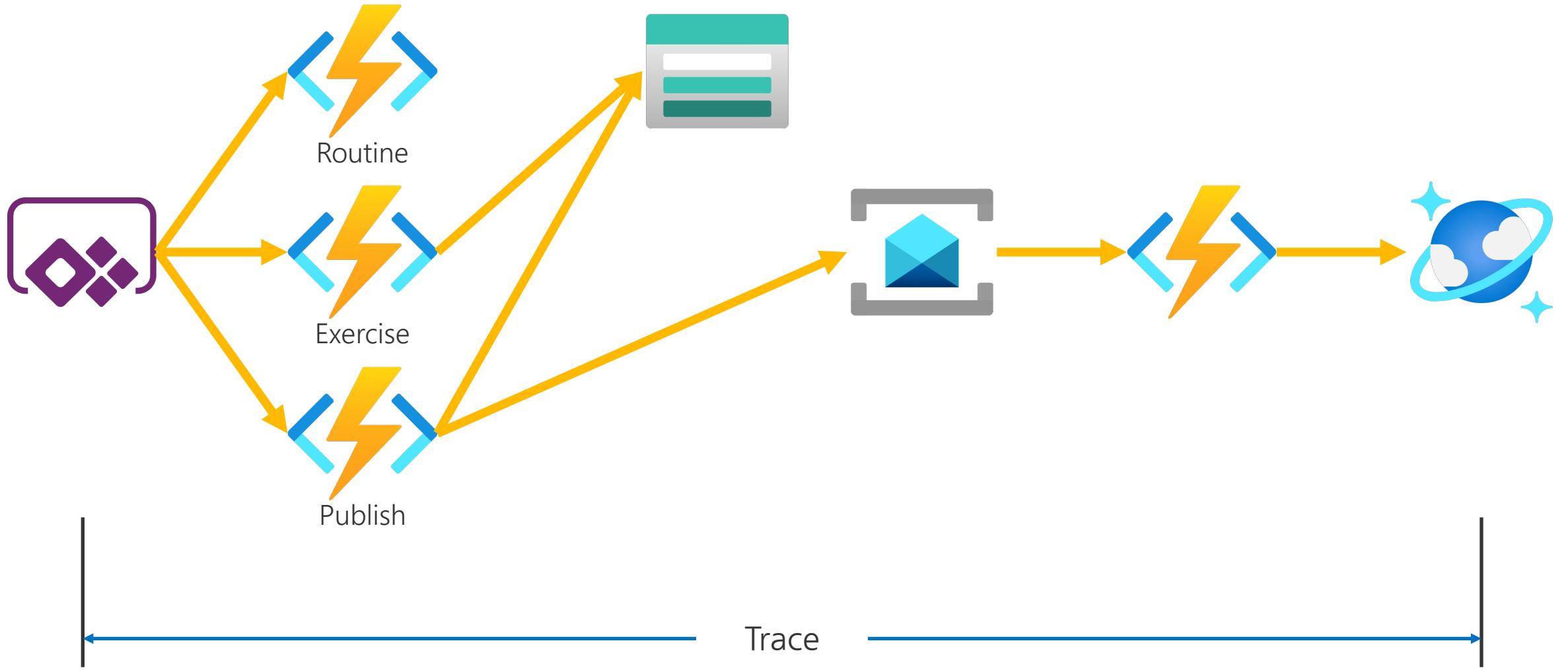
Telemetry Specification



샌드박스 프로젝트

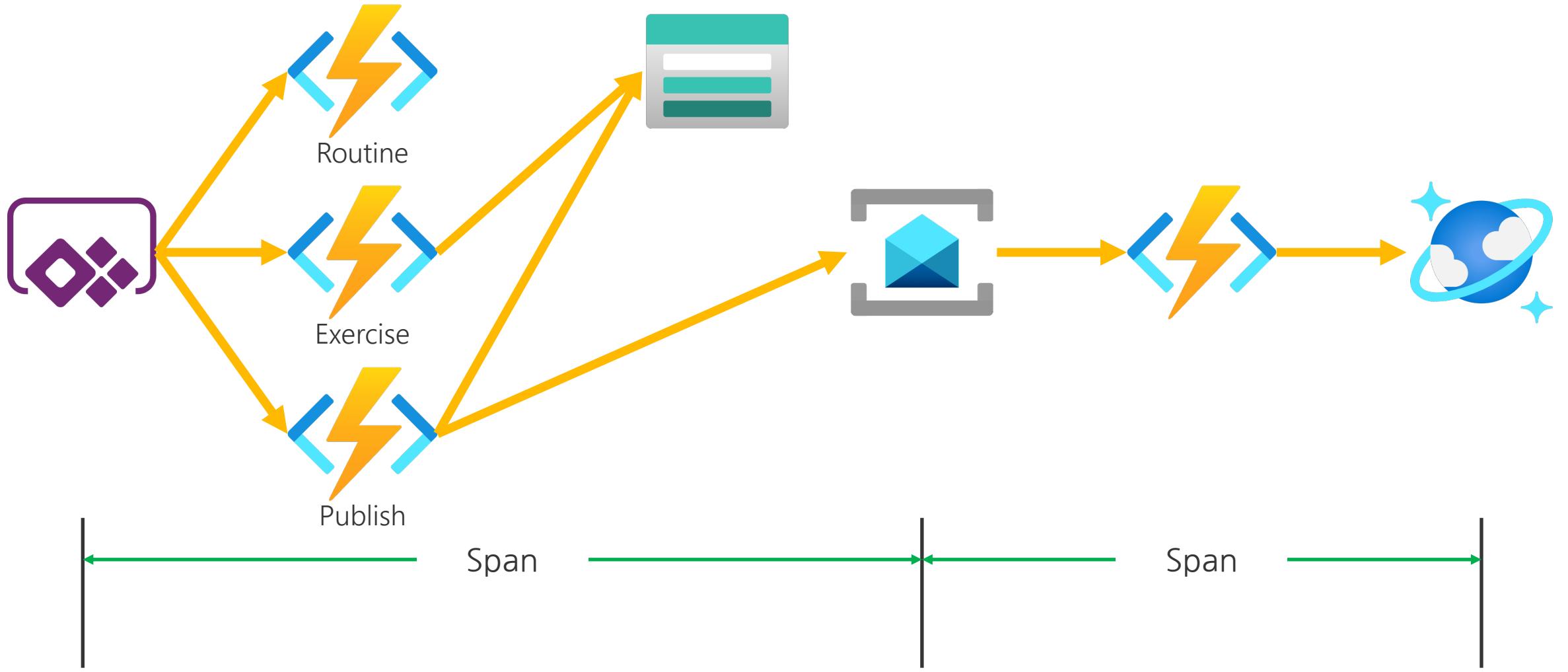
트레이스
Trace

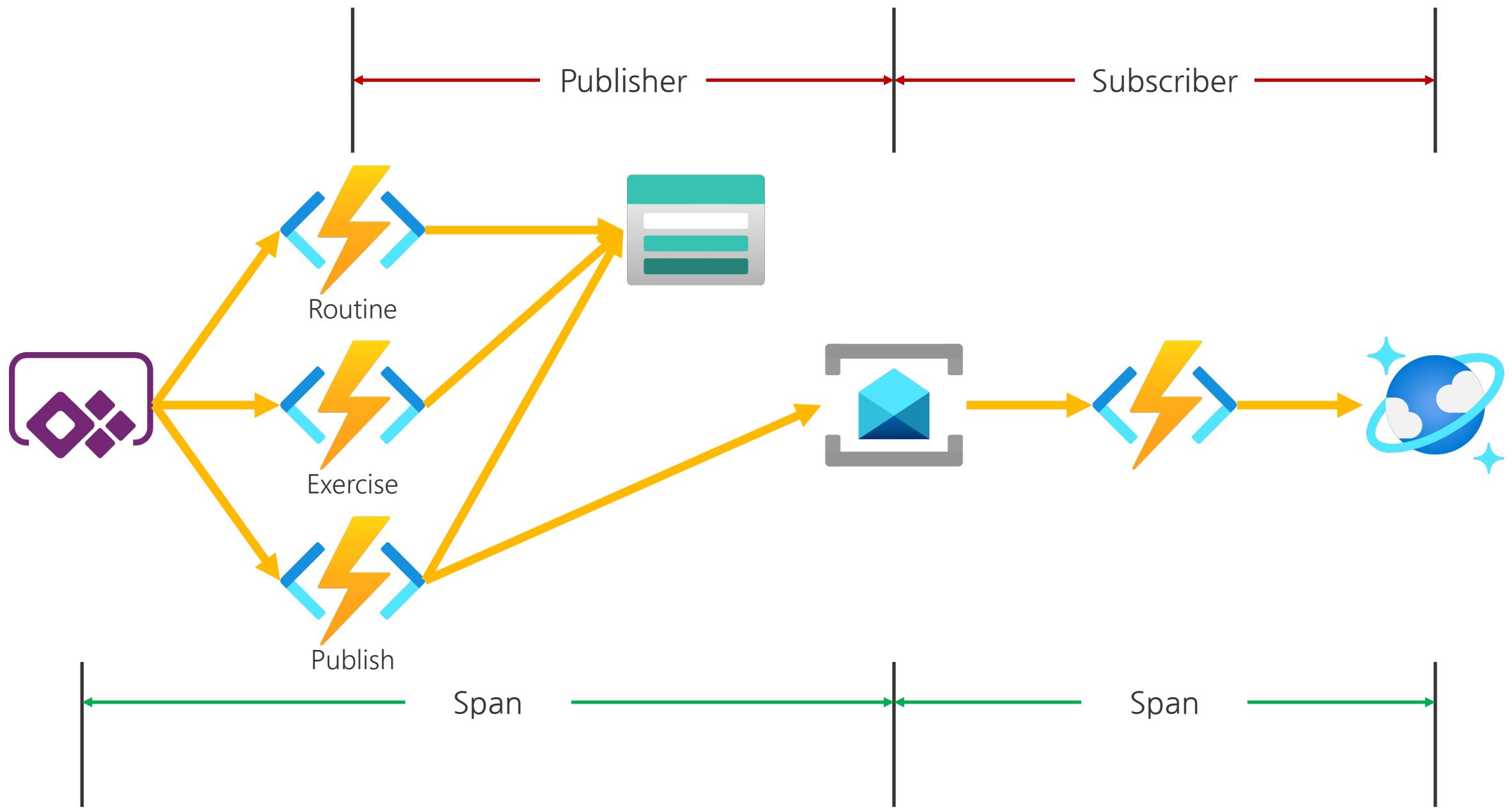
전체 메시지의 흐름



스팬
Span

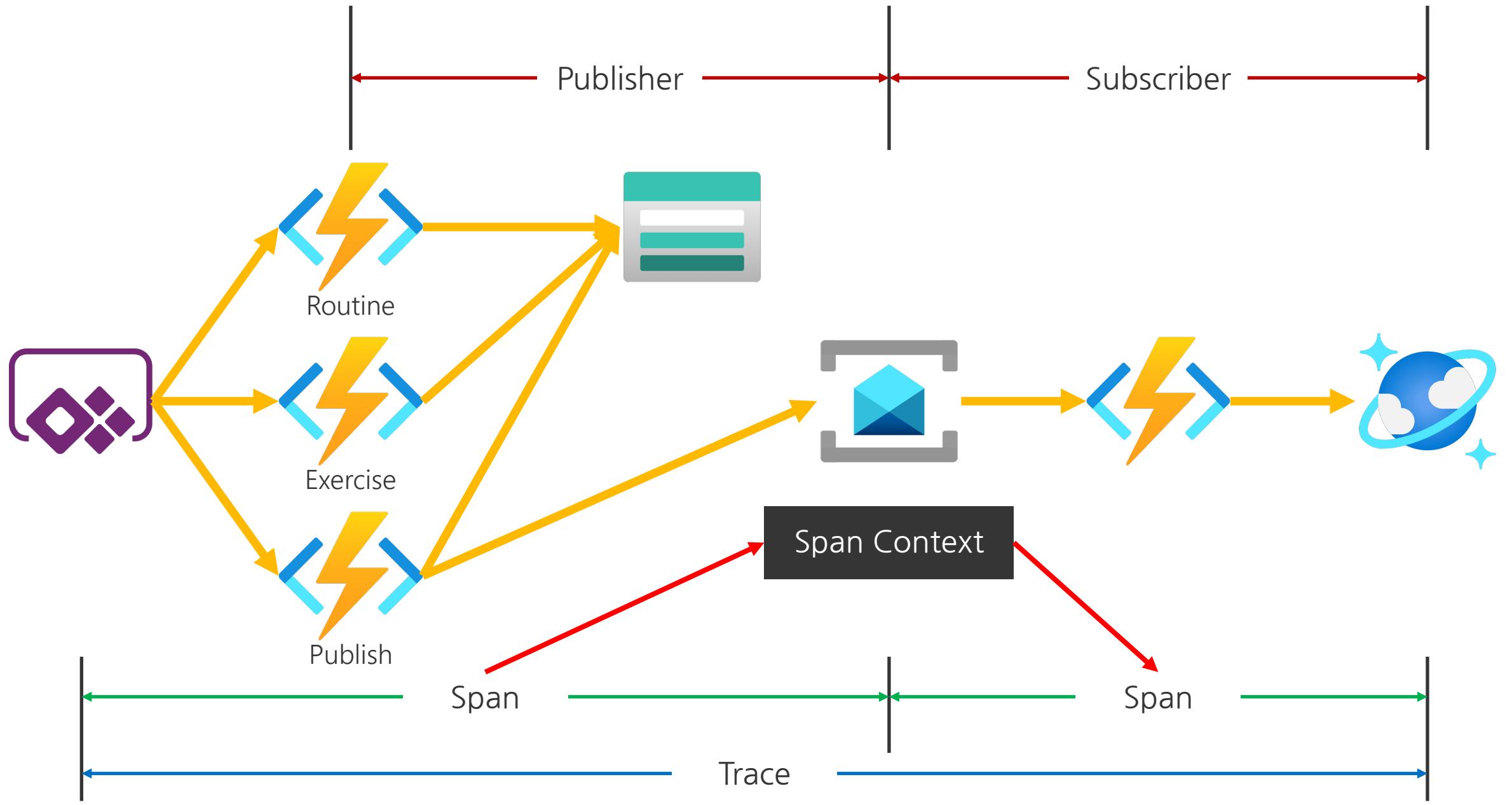
작업 단위

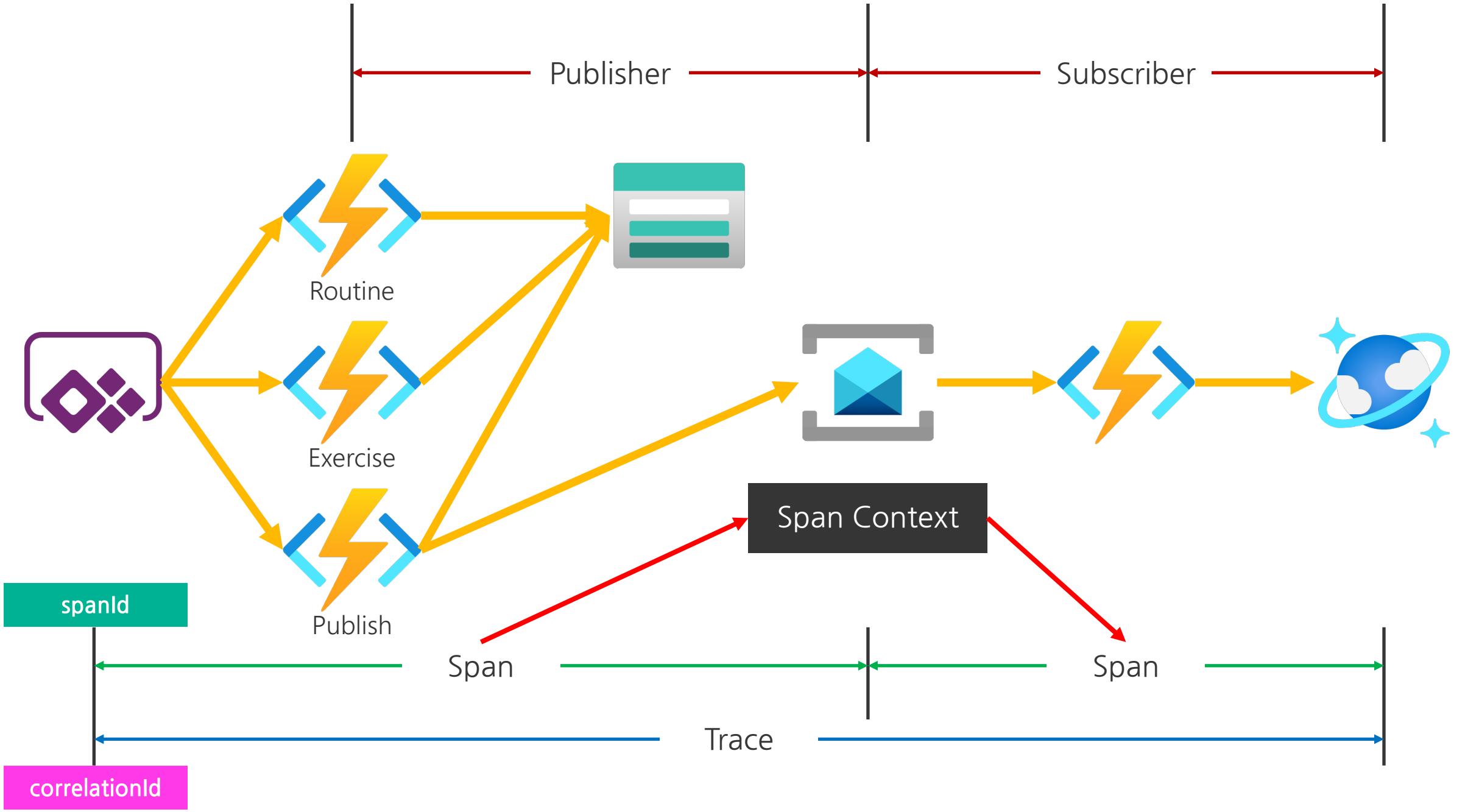


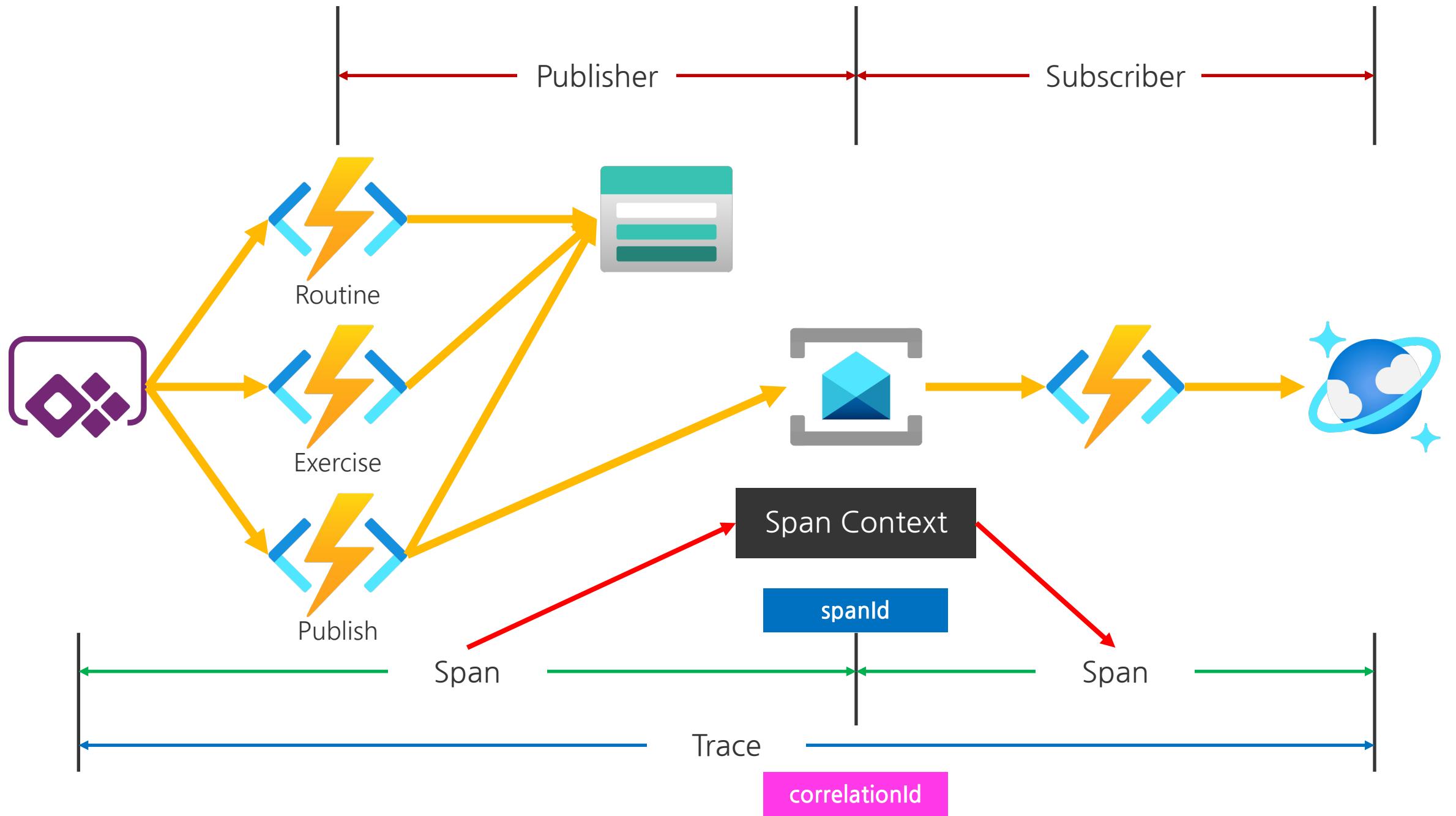


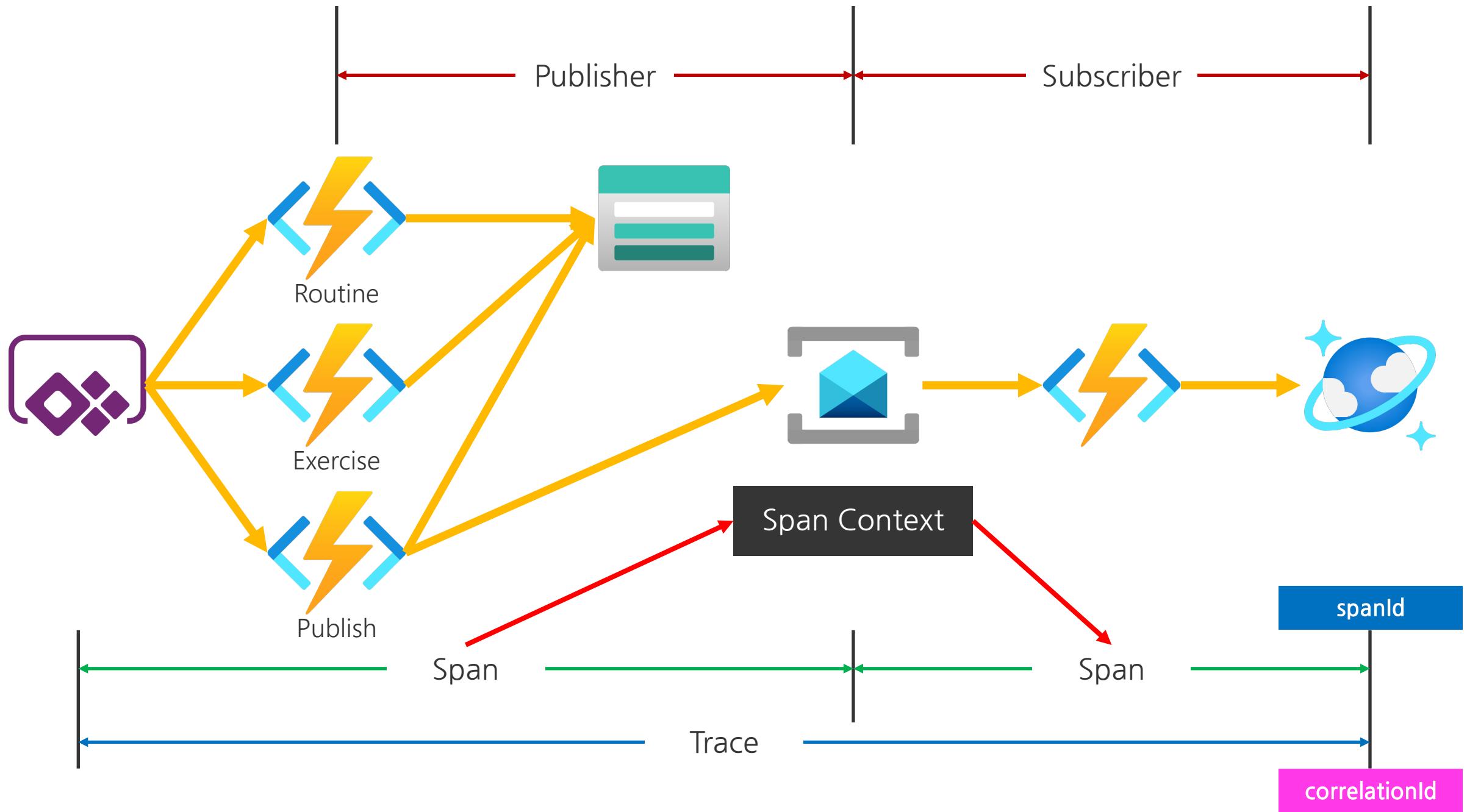
스팬 컨텍스트
Span Context

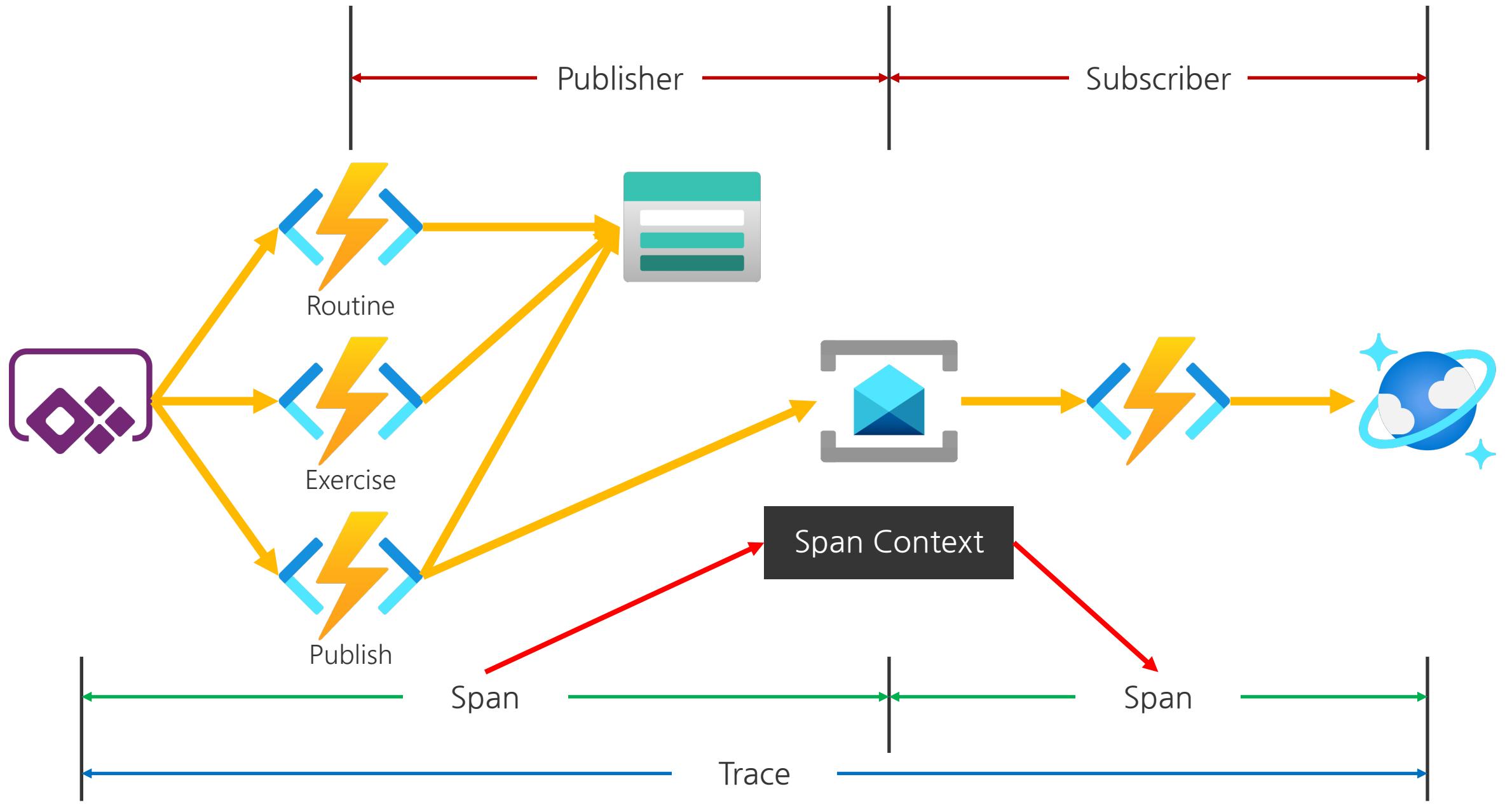
메타 데이터













Documentation

[API Reference](#)

[Code Examples](#)

C++

.NET

Erlang/Elixir

Go

Java

Javascript

PHP

Python

Ruby

Rust

Swift

다양한 구현체



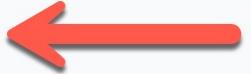
OPENTELEMETRY

Documentation



C++

.NET



Erlang/Elixir

Go

Java

Javascript

PHP

Python

Ruby

Rust

Swift

OpenTelemetry.NET

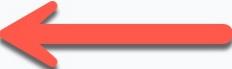


Documentation

...

...

C++

.NET 

Erlang/Elixir

Go

Java

Javascript

PHP

Python

Ruby

Rust

Swift

OpenTelemetry.NET

+

애플리케이션 인사이트



그런데 말입니다

애저 평션 지원 안함



추적용 체크포인트 직접 구현

사용자 지정 텔레메트리 속성

이벤트 종류, 이벤트 처리 상태

이벤트 종류, 이벤트 처리 상태
스팬 종류, 스팬 처리 상태

이벤트 종류, 이벤트 처리 상태
스팬 종류, 스팬 처리 상태
인터페이스 종류

이벤트 종류, 이벤트 처리 상태
스팬 종류, 스팬 처리 상태
인터페이스 종류
코릴레이션 ID (추적 ID)

[New screen](#)[Theme](#)

Open Sans

24

[B](#)[/](#)[U](#)[abc](#)[A](#)[≡](#)[Fill](#)[Border](#)[Reorder](#)[Align](#)[Group](#)

OnSelect

[fx](#)

```
ClearCollect(correlationId, GUID());  
ClearCollect(spanId, GUID());
```

[Format text](#)[Remove formatting](#)

Tree view

Screens Components



btnMainExit

tglMainInterface

btnStart



tglRoutineInterface

btnRoutineExit

lblNewRoutine

txbRoutine

> glryTargets

btnStartRoutine

...

Core

Full Body

Leg

Shoulder

Start

```
public async Task<IActionResult> CreateRoutineAsync(
    [HttpTrigger(AuthorizationLevel.Function, HttpVerbs.Post, Route = "routines")] HttpRequest req,
    ExecutionContext context,
    ILogger log)
{
    var request = await req.ToRequestMessageAsync<RoutineRequestMessage>().ConfigureAwait(false);
    var @interface = request.Interface;

    var correlationId = request.CorrelationId;
    var spanId = request.SpanId;

    var eventId = context.InvocationId;

    log.LogData(LogLevel.Information, request,
        EventType.RoutineReceived, EventStatusType.Succeeded, eventId,
        SpanType.Publisher, SpanStatusType.PublisherInitiated, spanId,
        @interface, correlationId);

    ...
}
```

```
public async Task<IActionResult> CreateRoutineAsync(  
    [HttpTrigger(AuthorizationLevel.Function, HttpVerbs.Post, Route = "routines")] HttpRequest req,  
    ExecutionContext context,  
    ILogger log)  
{  
    var request = await req.ToRequestMessageAsync<RoutineRequestMessage>().ConfigureAwait(false);  
    var @interface = request.Interface;  
  
    var correlationId = request.CorrelationId;  
    var spanId = request.SpanId;  
  
    var eventId = context.InvocationId;  
  
    log.LogData(LogLevel.Information, request,  
        EventType.RoutineReceived, EventStatusType.Succeeded, eventId,  
        SpanType.Publisher, SpanStatusType.PublisherInitiated, spanId,  
        @interface, correlationId);  
    ...  
}
```

```
var messageId = Guid.NewGuid();
var subSpanId = Guid.NewGuid();
var timestamp = DateTimeOffset.UtcNow;
var message = (RoutineQueueMessage)(PublishResponseMessage)res.Value;
var msg = new ServiceBusMessage(messageToJson())
{
    CorrelationId = correlationId.ToString(),
    MessageId = messageId.ToString(),
    ContentType = ContentTypes.ApplicationJson,
};

msg.ApplicationProperties.Add("pubSpanId", spanId);
msg.ApplicationProperties.Add("subSpanId", subSpanId);
msg.ApplicationProperties.Add("interface", @interface.ToString());
msg.ApplicationProperties.Add("timestamp", timestamp.ToString(CultureInfo.InvariantCulture));

await collector.AddAsync(msg).ConfigureAwait(false);

log.LogData(LogLevel.Information, msg,
    EventType.MessagePublished, EventStatusType.Succeeded, eventId,
    SpanType.Publisher, SpanStatusType.PublisherInProgress, spanId,
    @interface, correlationId,
    messageId: messageId.ToString(),
    message: EventType.MessagePublished.ToDisplayName());
```

```
var messageId = Guid.NewGuid();
var subSpanId = Guid.NewGuid();
var timestamp = DateTimeOffset.UtcNow;
var message = (RoutineQueueMessage)(PublishResponseMessage)res.Value;
var msg = new ServiceBusMessage(messageToJson())
{
    CorrelationId = correlationId.ToString(),
    MessageId = messageId.ToString(),
    ContentType = ContentTypes.ApplicationJson,
};

msg.ApplicationProperties.Add("pubSpanId", spanId);
msg.ApplicationProperties.Add("subSpanId", subSpanId);
msg.ApplicationProperties.Add("interface", @interface.ToString());
msg.ApplicationProperties.Add("timestamp", timestamp.ToString(CultureInfo.InvariantCulture));

await collector.AddAsync(msg).ConfigureAwait(false);

log.LogData(LogLevel.Information, msg,
    EventType.MessagePublished, EventStatusType.Succeeded, eventId,
    SpanType.Publisher, SpanStatusType.PublisherInProgress, spanId,
    @interface, correlationId,
    messageId: messageId.ToString(),
    message: EventType.MessagePublished.ToDisplayName());
```

```
public async Task IngestAsync(
    [ServiceBusTrigger(GymLogTopicKey, GymLogSubscriptionKey)] ServiceBusReceivedMessage msg,
    ExecutionContext context,
    ILogger log)
{
    ...
    var @interface = Enum.Parse<InterfaceType>(msg.ApplicationProperties["interface"] as string, ignoreCase: true);

    var correlationId = Guid.Parse(msg.CorrelationId);
    var spanId = (Guid)msg.ApplicationProperties["subSpanId"];

    var messageId = Guid.Parse(msg.MessageId);
    var eventId = context.InvocationId;

    log.LogData(LogLevel.Information, message,
        EventType.MessageReceived, EventStatusType.Succeeded, eventId,
        SpanType.Subscriber, SpanStatusType.SubscriberInitiated, spanId,
        @interface, correlationId);
}
```

```
public async Task IngestAsync(
    [ServiceBusTrigger(GymLogTopicKey, GymLogSubscriptionKey)] ServiceBusReceivedMessage msg,
    ExecutionContext context,
    ILogger log)
{
    ...
    var @interface = Enum.Parse<InterfaceType>(msg.ApplicationProperties["interface"] as string, ignoreCase: true);

    var correlationId = Guid.Parse(msg.CorrelationId);
    var spanId = (Guid)msg.ApplicationProperties["subSpanId"];

    var messageId = Guid.Parse(msg.MessageId);
    var eventId = context.InvocationId;

    log.LogData(LogLevel.Information, message,
        EventType.MessageReceived, EventStatusType.Succeeded, eventId,
        SpanType.Subscriber, SpanStatusType.SubscriberInitiated, spanId,
        @interface, correlationId);
}
```

Oops!

Check with Admin:

* Correlation ID: 5380912f-9c8f-466c-a78c-e5fe194bd21f

[Back](#)



appins-gymlog-dev-krc | Logs



Application Insights

New Query 1*



appins-gymlog-de...

Select scope

Run

Time range: Last 24 hours

Save

Share

New alert rule

Export

Pin to dashboard

Format query

Feedback

Queries

Query explorer



```
1 let correlationId = "5380912f-9c8f-466c-a78c-e5fe194bd21f";
2 traces
3 | sort by timestamp desc
4 | where customDimensions.prop_correlationId == correlationId
5 | project Timestamp = timestamp
6 ..... , LogLevel = customDimensions.prop_logLevel
7 ..... , CorrelationId = tostring(customDimensions.prop_correlationId)
8 ..... , Interface = customDimensions.prop_interfaceType
9 ..... , SpanType = customDimensions.prop_spanType
10 ..... , SpanStatus = customDimensions.prop_spanStatus
11 ..... , SpanId = tostring(customDimensions.prop_spanId)
12 ..... , EventType = customDimensions.prop_eventType
13 ..... , EventStatus = customDimensions.prop_eventStatus
14 ..... , EventId = tostring(customDimensions.prop_eventId)
15 ..... , EntityType = customDimensions.prop_entityType
```

Results Chart | Columns ▾ | Display time (UTC+00:00) ▾ Group columns

Completed. Showing results from the last 24 hours.

00:00.2 4 records

Schema and Filter

Timestamp [UTC]	CorrelationId	SpanType	SpanStatus	EventType	EventStatus	EntityType	MessageId	RecordId
> 5/18/2021, 3:22:08.04...	5380912f-9c8f-466c-a78c-e5fe...	Publisher	PublisherInProgr...	ExerciseNotCre...	Failed	ErrorResponseMess...	(null)	(null)
> 5/18/2021, 3:22:08.04...	5380912f-9c8f-466c-a78c-e5fe...	Publisher	PublisherInProgr...	ExerciseReceived	Succeeded	ExerciseRequestMe...	(null)	(null)
> 5/18/2021, 3:21:44.292...	5380912f-9c8f-466c-a78c-e5fe...	Publisher	PublisherInProgr...	RoutineCreated	Succeeded	RoutineResponseM...	(null)	(null)
> 5/18/2021, 3:21:44.168...	5380912f-9c8f-466c-a78c-e5fe...	Publisher	PublisherInitiated	RoutineReceived	Succeeded	RoutineRequestMes...	(null)	(null)



Page

1

of 1



50

▼

items per page

1 - 4 of 4 items



appins-gymlog-dev-krc | Logs

Application Insights

New Query 1*



appins-gymlog-de...

Select scope



Time range : Last 7 days



Save



Share



New alert rule



Export



Pin to dashboard

Format query

Feedback

Queries

Query explorer

Format query

```
1 let correlationId = "21a4821e-01b3-45d5-ba08-d36609b84de4";
2 traces
3 | sort by timestamp desc
4 | where customDimensions.prop__correlationId == correlationId
5 | project Timestamp = timestamp
6 , LogLevel = customDimensions.prop__logLevel
7 , CorrelationId = tostring(customDimensions.prop__correlationId)
8 , Interface = customDimensions.prop__interfaceType
9 , SpanType = customDimensions.prop__spanType
```



Results

Chart



Display time (UTC+09:00)



Group columns

Completed. Showing results from the last 7 days.

00:00.2 26 records

Schema and Filter

e...	CorrelationId	SpanType	SpanStatus	EventType	EventStatus	EntityType	MessageId	RecordId
4/20...	21a4821e-01b3-45...	Subscriber	SubscriberCompl...	MessageProcessed	Succeeded	RoutineQueueMessage	(null)	9c5872ab-c140-494a-8e96-fe25...
4/20...	21a4821e-01b3-45...	Subscriber	SubscriberInitiated	MessageReceived	Succeeded	RoutineQueueMessage	(null)	(null)
4/20...	21a4821e-01b3-45...	Publisher	PublisherComple...	RoutineCompleted	Succeeded	RoutineResponseMe...	(null)	(null)
4/20...	21a4821e-01b3-45...	Publisher	PublisherInProgr...	MessagePublished	Succeeded	ServiceBusMessage	9c5872ab-c140-494a-8e96-fe25...	(null)
4/20...	21a4821e-01b3-45...	Publisher	PublisherInProgr...	RecordPopulated	Succeeded	PublishResponseMes...	(null)	(null)
4/20...	21a4821e-01b3-45...	Publisher	PublisherInProgr...	PublishRequestRec...	Succeeded	PublishRequestMess...	(null)	(null)
4/20...	21a4821e-01b3-45...	Publisher	PublisherInProgr...	ExerciseCreated	Succeeded	ExerciseResponseMe...	(null)	(null)
4/20...	21a4821e-01b3-45...	Publisher	PublisherInProgr...	ExerciseReceived	Succeeded	ExerciseRequestMes...	(null)	(null)



Page

1

of 1



50



items per page

1 - 26 of 26 items

데모: 종단간 메시지 추적

정리

가상의 헬스장

가상의 헬스장 파워 앱 개발

가상의 헬스장
파워 앱 개발
오픈 텔레메트리

가상의 헬스장
파워 앱 개발
오픈 텔레메트리
파워 앱 + 오픈 텔레메트리

가상의 헬스장
파워 앱 개발
오픈 텔레메트리
파워 앱 + 오픈 텔레메트리
메시지 추적

블로그 포스트
aka.ms/skug/tracing



소스 코드
aka.ms/skug/gymlog



전문가가 추천하는 가이드

Microsoft Docs

기술 소개, 튜토리얼,
코드 샘플 등
다양한 기능을 살펴보세요.

aka.ms/skug/docs

The screenshot shows the Microsoft Docs homepage. At the top, there's a navigation bar with the Microsoft logo, 'Docs', and links for '설명서', 'Learn', 'Q&A', and '코드 샘플'. Below the header, the URL 'docs.microsoft.com' is displayed, followed by a sub-headline: '개발자와 기술 전문가용 Microsoft 설명서 및 학습을 위한 품입니다.' A search bar with placeholder text '문서, 교육 및 코드 샘플 검색...' and a blue '검색' button are visible. The main content area features three main categories: '설명서' (Documentation), 'Learn', and 'Q&A'. Each category has an icon and a brief description. The '설명서' section says 'Microsoft 제품의 방대한 설명서 리소스를 검색합니다.' The 'Learn' section says '역할, 실무 또는 기술 별로 학습 경로를 살펴봅니다.' The 'Q&A' section says '기술 커뮤니티의 다른 사용자에게 질문하여 그들로부터 배웁니다.' At the bottom, there's a section titled '추천 Microsoft Learn' with a sub-headline: '초보 사용자이든, 경험 많은 전문가이든 관계없이 Microsoft의 실습 방식은 더 자신감을 가지고 자신에게 맞는 속도로 더 빠르게 목표에 도달할 수 있게 도와줍니다.' To the right, there's a small cartoon character holding a smartphone.

언제 어디서나 배워보세요

Microsoft Learn

단계별 학습을 통해
다양한 기능을 알아보고
새로운 기술을 배워보세요.

aka.ms/skug/learn

The screenshot shows the Microsoft Learn website with a dark-themed header and navigation bar. The main content area features a large black box with white text. The text reads: "MICROSOFT LEARN에 오신 것을 환영합니다." (Welcome to Microsoft Learn), "경로 살펴보기" (Explore Path), and "초보 사용자이든, 경험 많은 전문가이든 관계없이 Microsoft의 실습 방식은 더 자신감을 가지고 자신에게 맞는 속도로 더 빠르게 목표에 도달할 수 있게 도와줍니다." (Whether you're a beginner or an experienced professional, Microsoft's learning approach helps you reach your goals faster by providing confidence and speed). Below this box is a blue button labeled "내 경로 사용자 지정" (Customize my path). To the right of the main content, there are several cards with icons and text: "학습 경로" (Learning Path) with the subtext "자신에게 맞는 일정으로 학습" (Learn at your own pace) and a description about learning paths; "인증" (Certification) with the subtext "Microsoft 인증받기" (Get certified) and a description about certifications; and a card with a TV icon and text. The background of the slide features abstract blue, yellow, and red curved lines on a light gray grid pattern.

질문 있으신 분?





감사합니다!

파워 앱과 서비스로
홈 트레이닝 앱
로그 데이터 추적하기

