Microsoft

Supply chain integration: Leveraging an Azure event-based architecture from SAP

Kevin Wilson

Agenda

- Problem statement
- Solution overview
- High-level architecture
- Solution benefits
- Way forward

Microsoft

Problem statement

Critical issues:

External business processes

- No **real-time** access to SAP events (status change of a business object)
- No near **real-time** monitoring → no real time attention **and** response

Long lead time to deploy

- Engineering / IT resources needed to “code” for each new scenario
- SAP is mission critical, and any development change follows strict security and compliance rules

Increased resource costs

- Frequent polling of SAP data by consuming apps to get latest object state

Examples:

Timely notification to suppliers informing them on a status change to a PO

Ensuring accruals are processed with the latest information about a PO

Early warning to updates and failures with ASNs and GRs

Solution goals and guiding principles

Build an SAP near real-time event publishing (NRTEP) service

SAP | ASAPIO IT Solutions & Services | Event Grid

Modern | Flexible | Agile | Efficient

Easy, scalable access to SAP data from outside SAP ecosystem

Isolate event consumers from SAP

Isolate SAP from demands of multiple consumers

Industry event standard

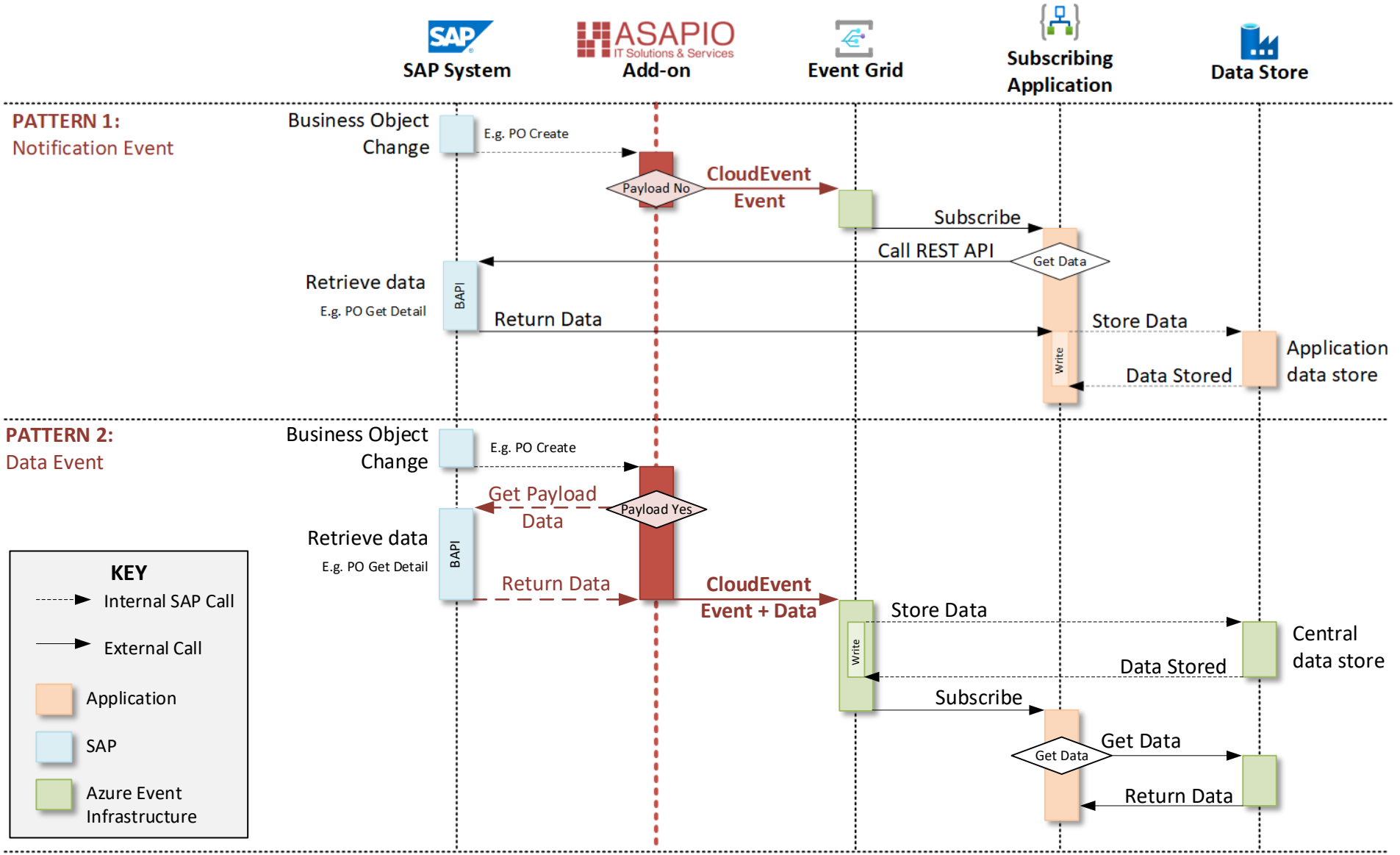
Cloud **event standard**

spec/spec.md at v1.0.1 · cloudevents/spec · GitHub

Azure first

Microsoft Azure **event grid**

Getting technical: understanding EBA event types



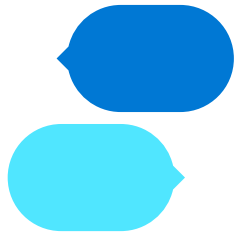
Notification event

```
{
  "specversion": "1.0",
  "type": "default.micrsoft.event.1.T1",
  "source": "SAP_EMS_INS/PO_CREATE",
  "id": "PO_CREATE20210106051719.7826060",
  "time": "2021-01-06T05:17:19.782Z",
  "data": {
    "objectType": "BUS2012",
    "event":     "CREATED",
    "objectId":   "4100030464",
    "userid":     "KEVWILS",
    ".....   ← Custom properties / attributes / filenames
  }
}
```

Event + data

Solution overview – pilot project

- My order scenario
  - Purchase order visibility and exception management / monitoring
    - Objects: POs, invoice and GR
  - ~60k PO changes / year
  - Internal notifications of PO changes and exceptions
  - What does success look like?
    - Reduction in the number of tickets raised for inquiring on PO status (measurable)
    - Most up-to-date state of a PO is automatically available to users (through My Order app)
    - Current turnaround time for PO updates reduced from 21hrs to seconds
    - Reduced / elimination of polling SAP data for updates – initially ~70k / hour



Component overview – pilot project

Publisher

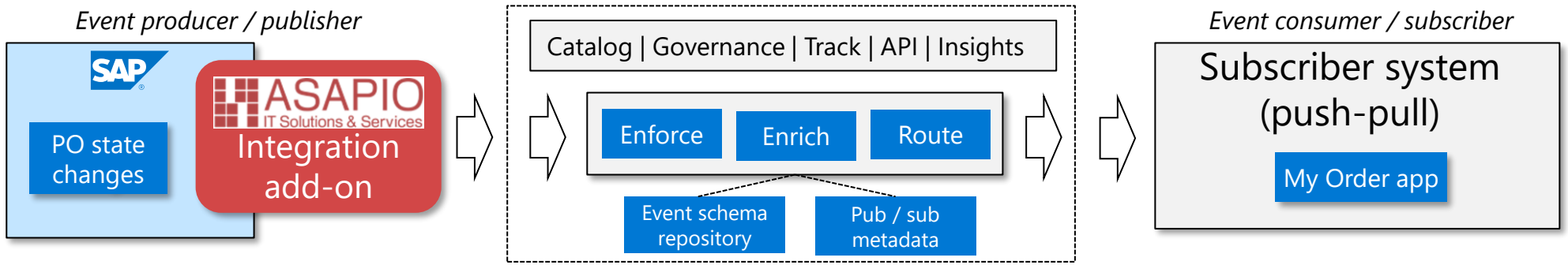
- Implement SAP ERP ASAPIO integration add-on
- Define and publish events
- Configure event schema

Platform

- Event catalog
- Cloud event schema
- Governance → activity audit, SOX compliance, Privacy / GDPR
- 100TPS throughput
- Enable push / pull model
- Telemetry, archive, reprocess / replay events

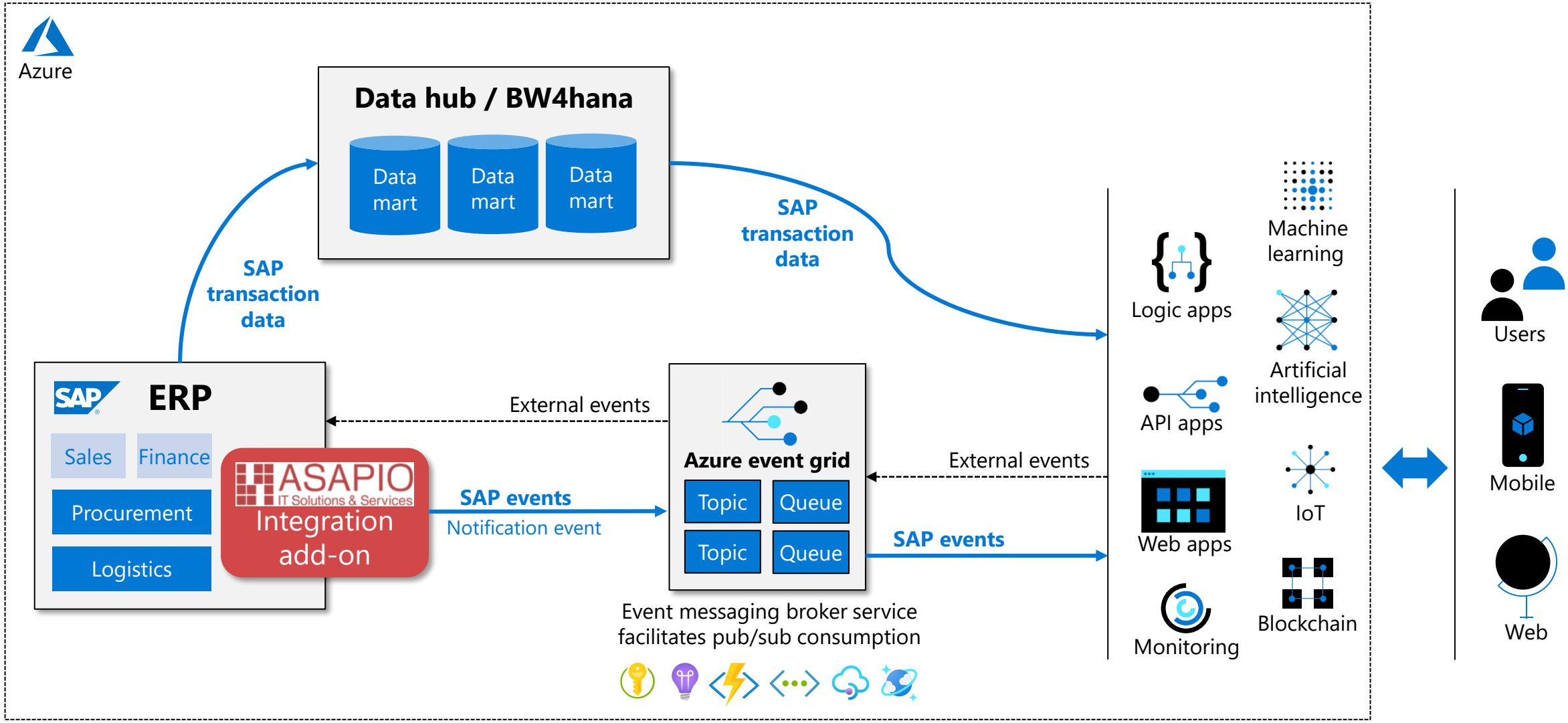
Subscriber

- Search and discover events
- Subscribe to events
- Configure subscription type and end point details
- Consume data within My Order app



Enabling event drive architecture

High-level architecture



Microsoft

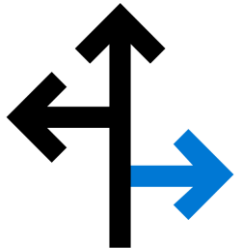
Solution benefits

Maximize

- Usage of data

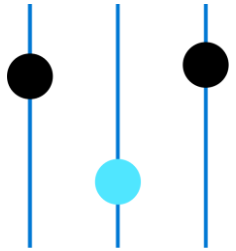
Efficient

- Event throughput
- Azure resource usage
- Real-time data



Flexible

- Multiple use cases



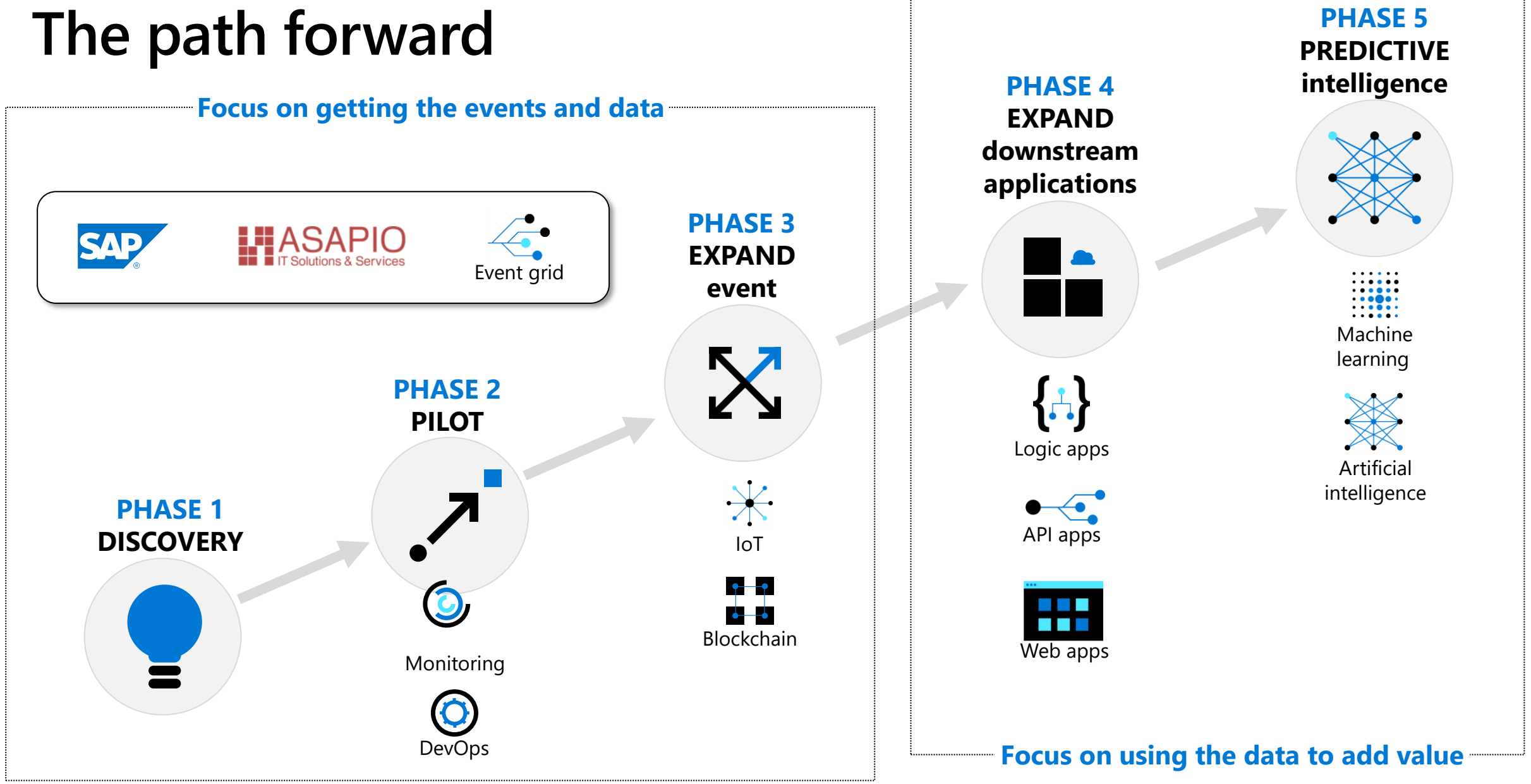
Agile

- Time-to-enable & use
- Time-to-insight
- Time-to-action

Scale

- Event throughput
- Volume handling

The path forward



Questions?

Kevin Wilson

Microsoft