OWASP CycloneDX - 1.5 Specification

Event-Trigger-Task Considerations

Matt Rutkowski, IBM
STSM, CTO Open Source Supply Chain Security

2022-10-24

Event-Drive Model - *Least Common Denominator (LCD)*

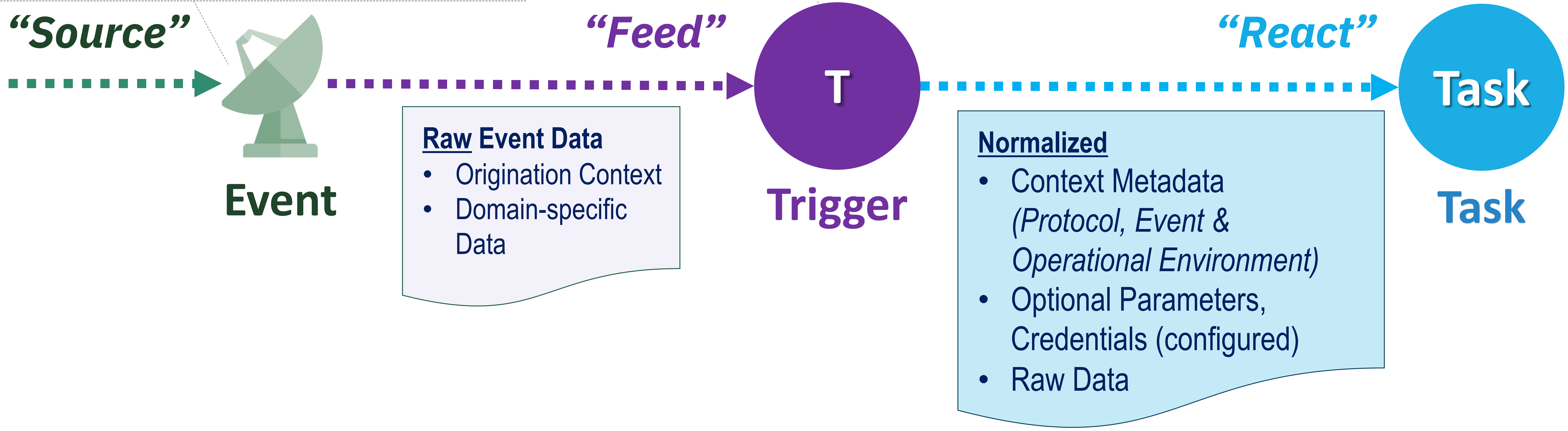
Events - representation of real-world, "***Source***" events that carry actionable Input data

- **Manual** (CLI) or **Automated** Events
- Carrying structured or unstructured data

Triggers - are named channels for a class of Events that "***Feed***" the Task

- Typically Named resources often backed by **Message Queues**
- **Potential Normalization of data**

Tasks - standalone functions invoked ***Reactively*** as an event handler



"Source"

"Feed"

"React"

Event

Trigger

Task

<u>Raw</u> Event Data

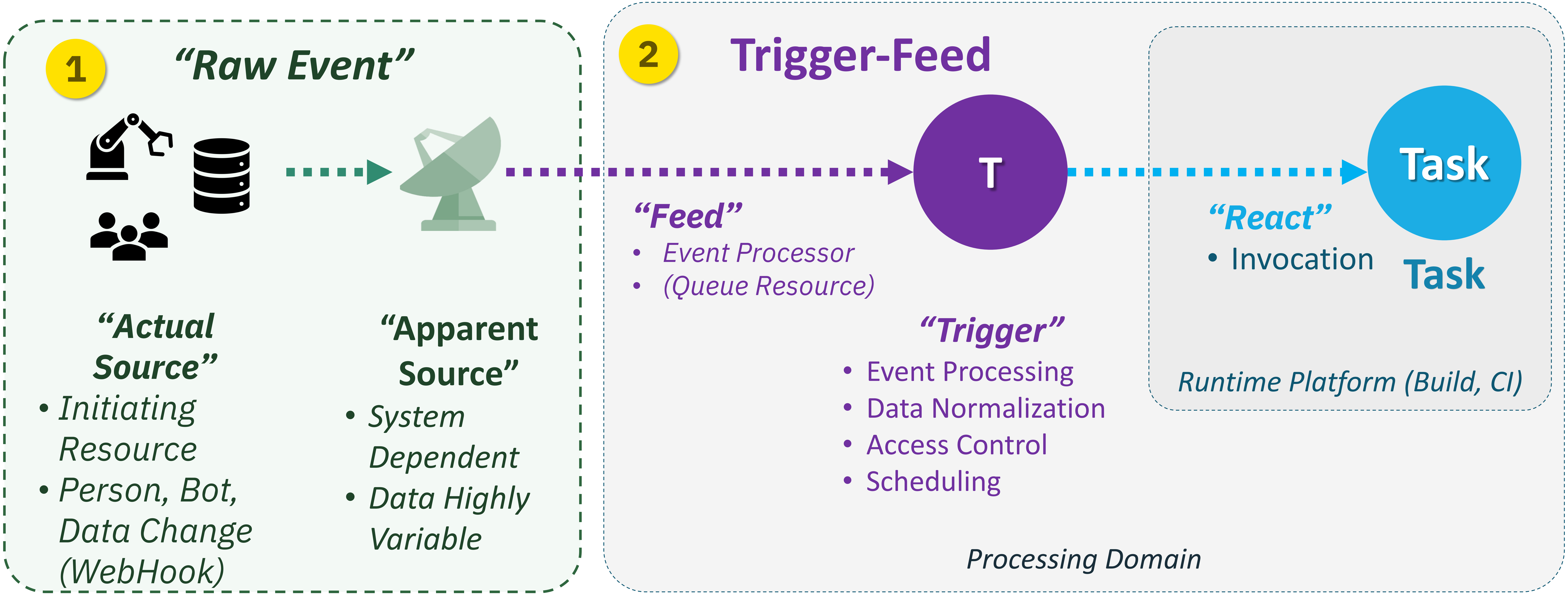
- Origination Context
- Domain-specific Data

<u>Normalized</u>

- Context Metadata *(Protocol, Event & Operational Environment)*
- Optional Parameters, Credentials (configured)
- Raw Data

Event-Drive Model – *Processing Chain*

Events are initiated by **Resources** and are Processed by **Triggers** which are associated to one or more **Tasks**



1 ***"Raw Event"***

"Actual Source"

- *Initiating Resource*
- *Person, Bot, Data Change (WebHook)*

"Apparent Source"

- *System Dependent*
- *Data Highly Variable*

2 **Trigger-Feed**

"Feed"

- *Event Processor*
- *(Queue Resource)*

T

"Trigger"

- Event Processing
- Data Normalization
- Access Control
- Scheduling

Task

"React"

- Invocation

Task

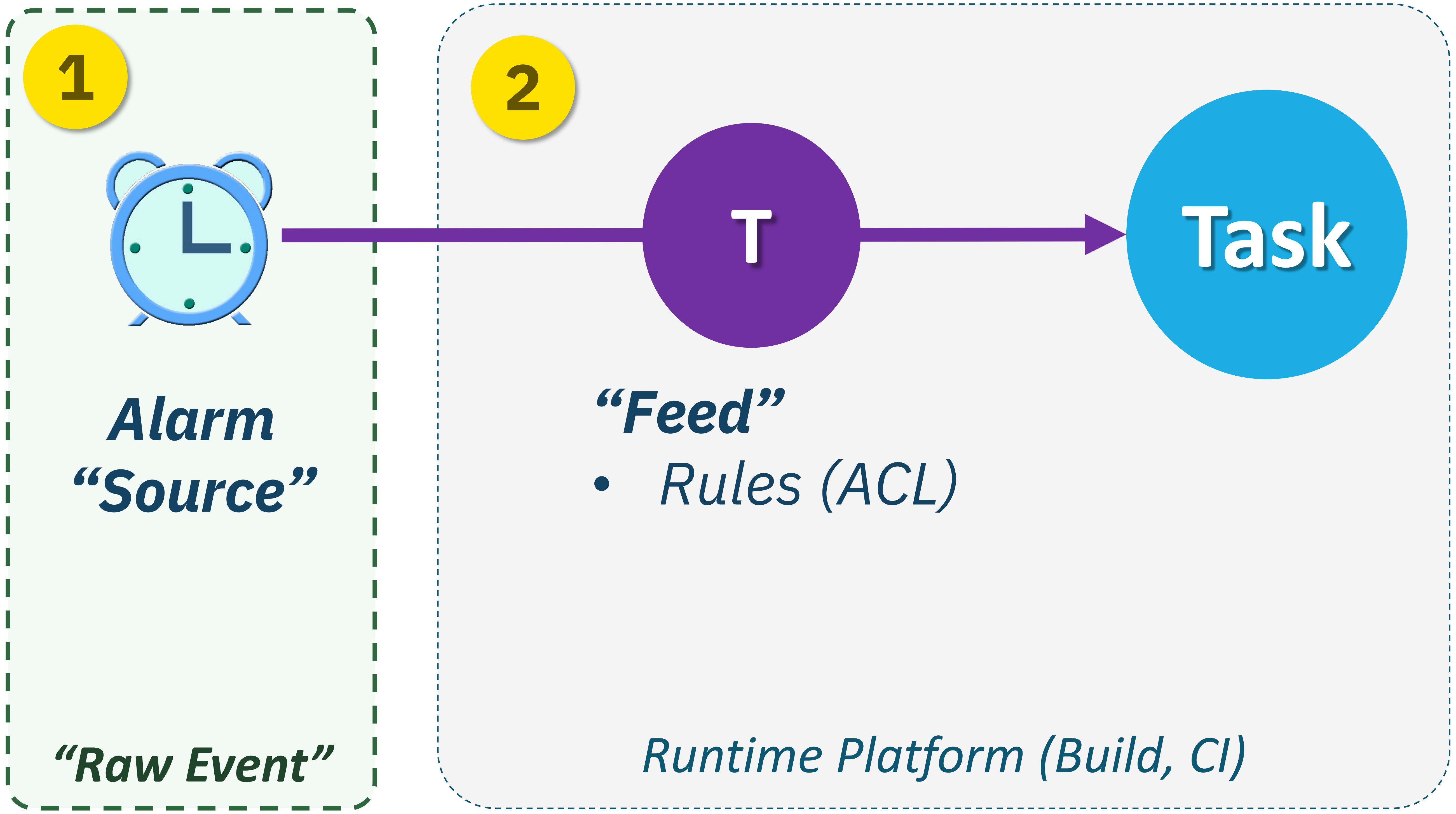
Runtime Platform (Build, CI)

Processing Domain

Scheduled Events - Periodic Tasks | "Cron Jobs"

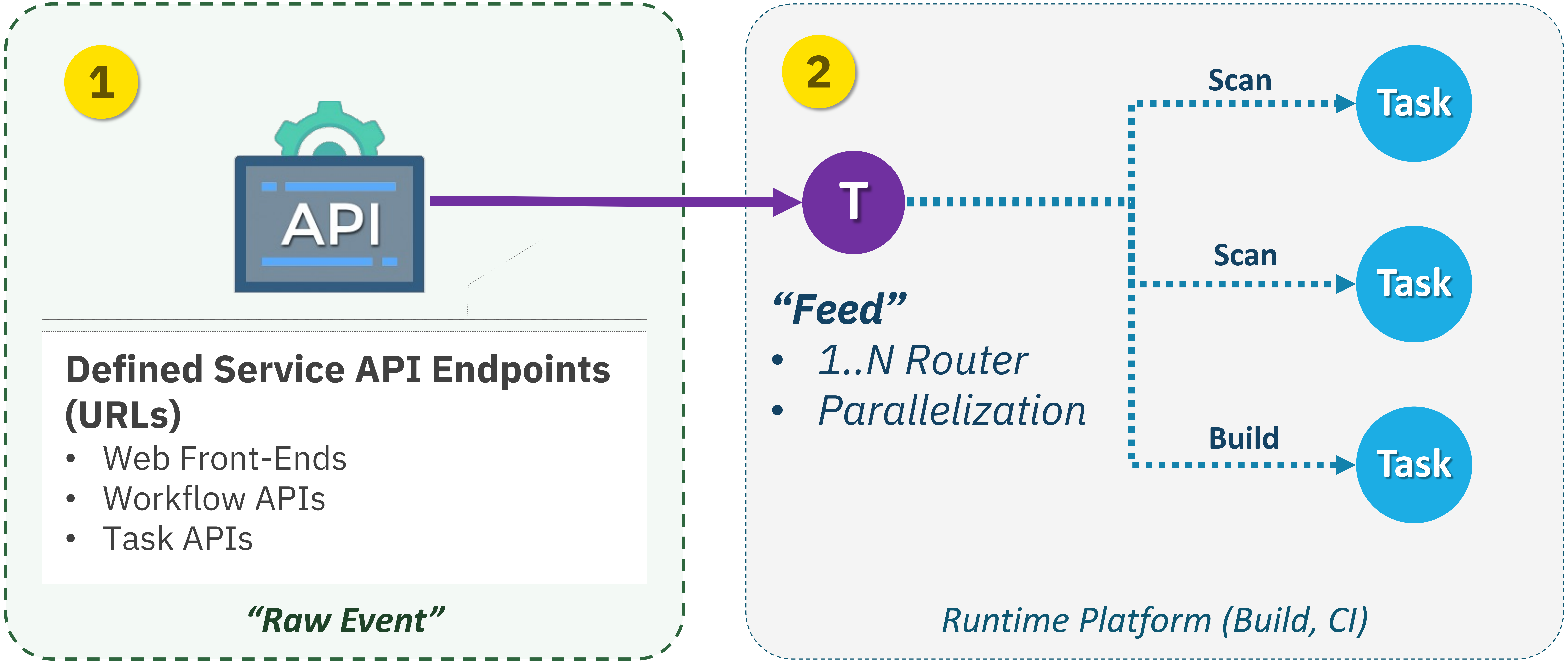
Considerations

- **Specific date/time**
  - Recurring or "fire once"
    - *Stage build every 2 weeks*
    - *Release on Jan. 1st 2023*
- **Periodic Intervals**
  - Run task every X mins/secs
    - *Scan code every 24 hours at 12 PM*
- **Time Windows** – (restricted)
  - Start / Stop by Date-Time
    - *Only execute Mon-Fri at 11AM to 2PM*



API Events *(explicit)*

Build/CI APIs invoked by Person (**Manual**) or Automated (i.e., **WebHooks**)



Defined Service API Endpoints (URLs)

- Web Front-Ends
- Workflow APIs
- Task APIs

"Raw Event"

"Feed"

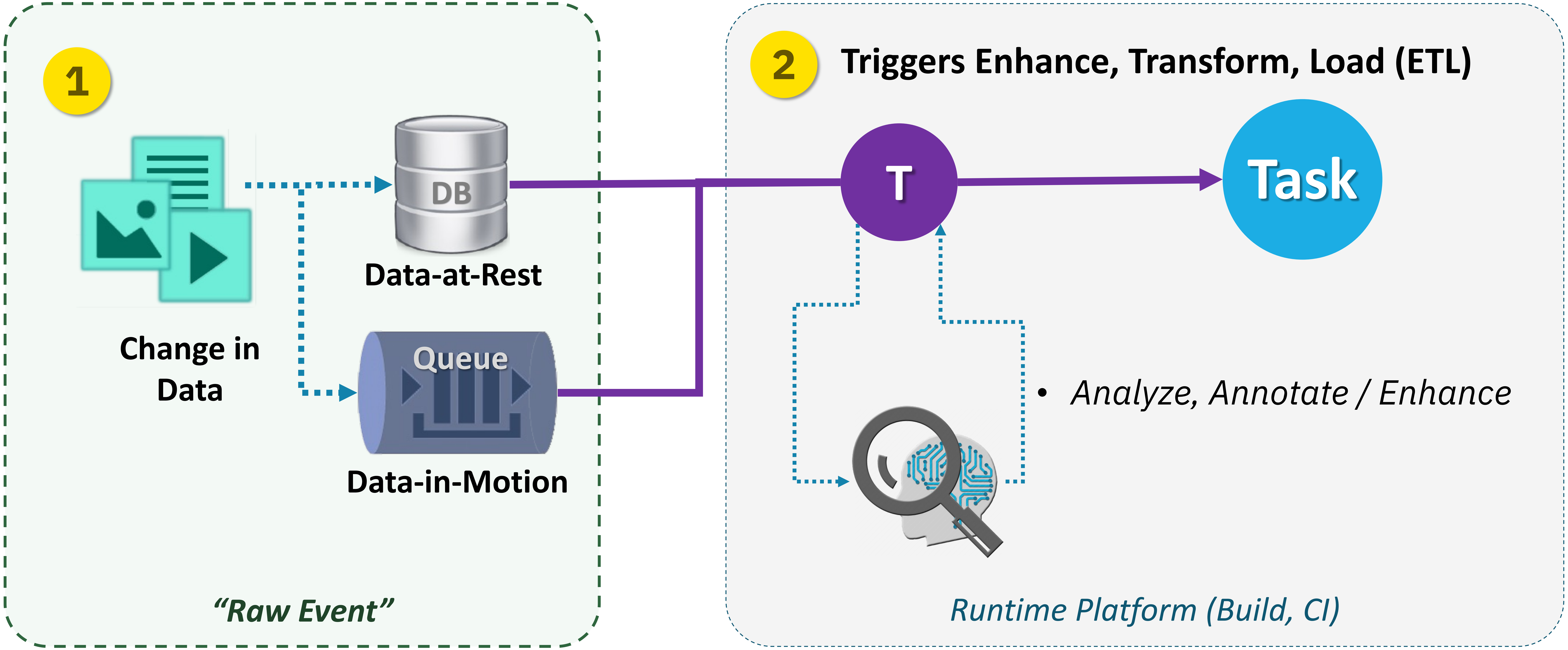
- *1..N Router*
- *Parallelization*

Runtime Platform (Build, CI)

Tasks may be on Disconnected Systems

Automated Events on Raw Data Changes *(implicit)*

Triggers are coupled to input data sources for ETL workloads


1
Change in Data
DB
Data-at-Rest
Queue
Data-in-Motion
"Raw Event"
2
Triggers Enhance, Transform, Load (ETL)
T
Task
• Analyze, Annotate / Enhance
Runtime Platform (Build, CI)

Includes GitHub events (e.g., Pull Request, Issue opened)

Security & Compliance Event Model

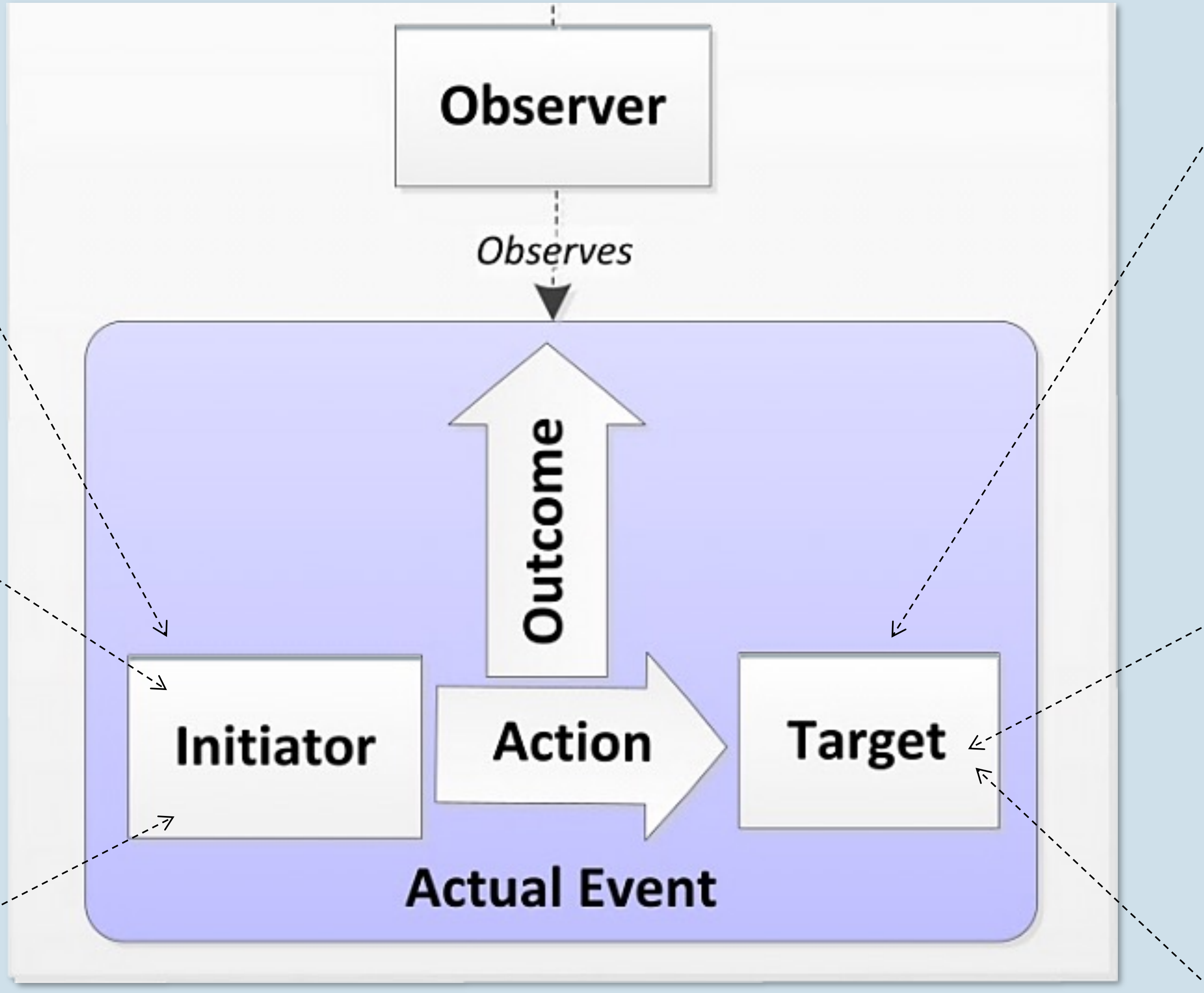
7 essential "W"s of Security and Compliance for Auditing

What
- What activity occurred?
- What was the result?

When
- When did the Action happen? When was it observed? How long did it take?
  - **ISO 8601 Timestamp with fractional sections (basic or precise) with Timezones (detailed)**
  - **NTP Server information**

Who (resource)
User / service that initiated the Action
- Initiator identifer, name (basic)
- Credentials (detailed)
- Identity assertions (precise)

*Events have an **optional** OBSERVER resource*



Where (resource)
- Resource where the event was "Consumed"
  - Observer identifer, name (basic)
  - Observer MAY also be the "Trigger" (Event Feed processor)

On What (resource)
- Resource did the Activity target
  - Target identifer, name (basic)
  - Universal Identifiers (detailed) (e.g., PURL)

FromWhere (resource)
- FromWhere was the Action Initiated?

ToWhere (resource)
- ToWhere was the Task (action) actually run?
  - Network addresses (basic)
  - Host information (agents, platforms, etc.) (detailed)
  - **ISO 6709 Geolocation**, ICANN codes (precise)

"Who: Includes GitHub Actions, Tekton (Cloud Events)