**1. Schedule / Tumbling / Event Grid Triggers**

**A. Schedule Trigger**

* **Definition**: Executes pipelines at specific times or intervals.
* **Use case**: Daily batch load, hourly reports, etc.
* **Configuration Fields**:
  + Start time (UTC)
  + Recurrence (Hourly, Daily, Weekly)
  + Time zone
* **Best practices**:
  + Use for regular, predictable intervals
  + Enable concurrency control for resource efficiency

**B. Tumbling Window Trigger**

* **Definition**: Trigger based on a periodic interval; each window is **non-overlapping** and **stateful**.
* **Use case**: Time-based partitioning of data (e.g., process logs per hour/day)
* **Key features**:
  + Late arrival tolerance
  + Dependency tracking
  + Can rerun failed windows
* **Best practices**:
  + Use for time-series or IoT-like data
  + Enable retry and concurrency settings for reliability

**C. Event Grid Trigger**

* **Definition**: Fires a pipeline when a specific event occurs (e.g., file dropped in Blob Storage).
* **Use case**: Real-time ingestion from external systems
* **Configuration**:
  + Event type: BlobCreated, BlobDeleted
  + Linked event subscription
* **Best practices**:
  + Combine with metadata filters to limit scope
  + Monitor for missed events using diagnostic logs

**2. Dependency Triggers**

* **Definition**: A trigger that activates a pipeline only after one or more other pipelines have successfully completed.
* **Use case**: Downstream workflows that depend on upstream data preparation
* **Key Elements**:
  + Dependent pipeline
  + Expected trigger window
  + Wait on dependency status (Success / Failed / Skipped)
* **Best practices**:
  + Use with tumbling windows to coordinate data batches
  + Avoid circular dependencies

**3. Exponential Back-off Retries**

* **Definition**: Retry mechanism where the wait time increases exponentially after each failure.
* **Use case**: Network glitches, transient failures, or throttled APIs
* **Configuration**:
  + Retry count (up to 5)
  + Retry interval
  + Exponential delay enabled (via code or integration runtime)
* **Formula**:

text

Delay = BaseInterval × 2^(RetryAttempt - 1)

* **Best practices**:
  + Enable for Copy Activities and REST/SOAP calls
  + Combine with alerting to notify after multiple failures

**4. Teams Webhook**

* **Definition**: A mechanism to post messages from ADF to Microsoft Teams channel via an HTTP webhook.
* **Use case**: Alert stakeholders on pipeline failures, SLA breaches, or data anomalies
* **Steps**:
  1. Create Incoming Webhook connector in Teams.
  2. Use Web Activity in ADF to POST to the webhook URL.
  3. Format payload as JSON (Adaptive Card format recommended).
* **Sample Payload**:

json

{

"text": "⚠️ ADF Pipeline Failed: PipelineName at @utcNow()"

}

* **Best practices**:
  1. Use parameterized messages for dynamic alerts
  2. Secure secrets in Azure Key Vault

**5. SLA Dashboard**

* **Definition**: A dashboard to track pipeline execution status vs expected completion time.
* **Use case**: Monitor and meet data delivery agreements (e.g., "All loads complete by 6 AM")
* **Components**:
  + ADF activity logs via Log Analytics
  + Kusto Query Language (KQL) queries
  + Power BI / Azure Monitor dashboards
* **Key Metrics**:
  + Pipeline start time, duration, success/failure
  + SLA breach flags (based on thresholds)
* **Best practices**:
  + Automate breach notification via Logic Apps or Webhooks
  + Maintain historical SLA logs for compliance

**Summary Table**

| **Feature** | **Use Case** | **Key Benefit** | **Tool/Method** |
| --- | --- | --- | --- |
| Schedule Trigger | Time-based loads | Predictable automation | Built-in ADF scheduler |
| Tumbling Trigger | Time-series ingestion | Stateful, windowed processing | Window trigger settings |
| Event Grid Trigger | Event-driven pipelines | Real-time responsiveness | Azure Event Grid |
| Dependency Trigger | Orchestrated execution flow | Task dependencies | Trigger chaining |
| Exponential Retry | Handle transient errors | Improved reliability | Retry settings in activity |
| Teams Webhook | Notifications | Real-time alerting | Web Activity + Webhook |
| SLA Dashboard | SLA tracking & alerting | Data governance, transparency | Log Analytics + Power BI |