

Recommendations

Solution Architecture

1. Components & Technologies

- **Data Sources:** User activity logs, content metadata, engagement history [stored in **Amazon S3 / Redshift**].
- **Baseline Recommendation Engine:** Rule-based or ML-based model for initial recommendations [**Python, Scikit-Learn, TensorFlow, or custom model**].
- **Amazon Personalize:** AWS service for machine learning-based recommendations.
- **Database:** **Amazon RDS [PostgreSQL / MySQL] or DynamoDB** to store recommendations.
- **Comparison & Reporting:** **Apache Spark / Pandas** for analysis and visualization.
- **Visualization & Insights:** **Amazon QuickSight / Grafana** for reporting.

Recommendation System Overview

Visualization & Insights

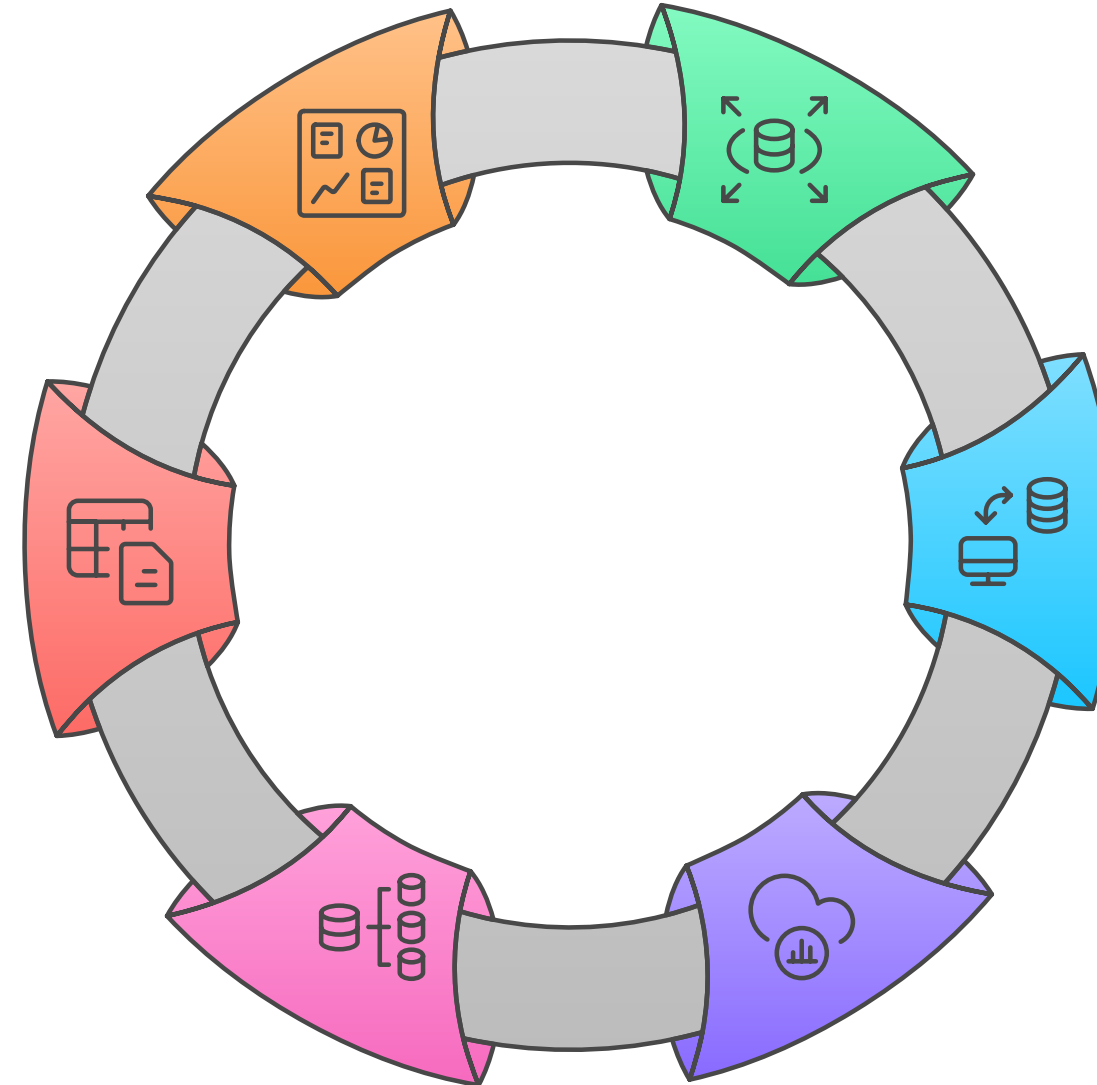
Generating reports and insights

Comparison & Reporting

Analyzing and visualizing recommendations

Database

Storing recommendations in RDS or DynamoDB



Data Sources

User activity and content data storage

Baseline Recommendation Engine

Initial recommendations using rules or ML

Amazon Personalize

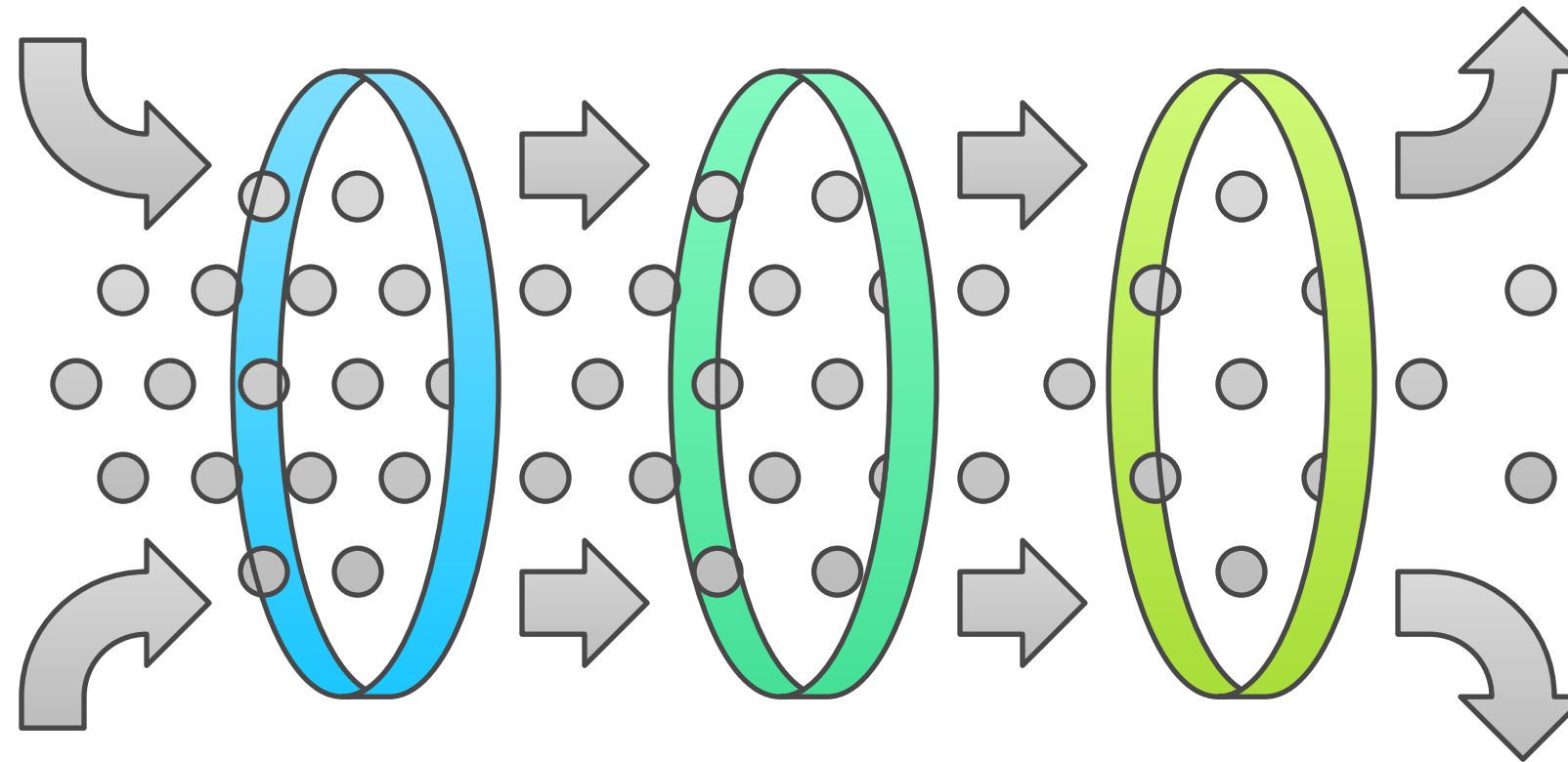
AWS service for advanced recommendations

2. Solution Workflow

Step 1: Data Collection

- Extract user data [**Paid + Monthly Active Free Users**] from **Redshift, MySQL, or DynamoDB**.
- Gather **content metadata** [Courses, Videos, Podcasts, etc.].
- Store raw data in **Amazon S3** for batch processing.

Data Collection and Storage Process



Extract User Data

Extract data from databases

Gather Content Metadata

Collect metadata from sources

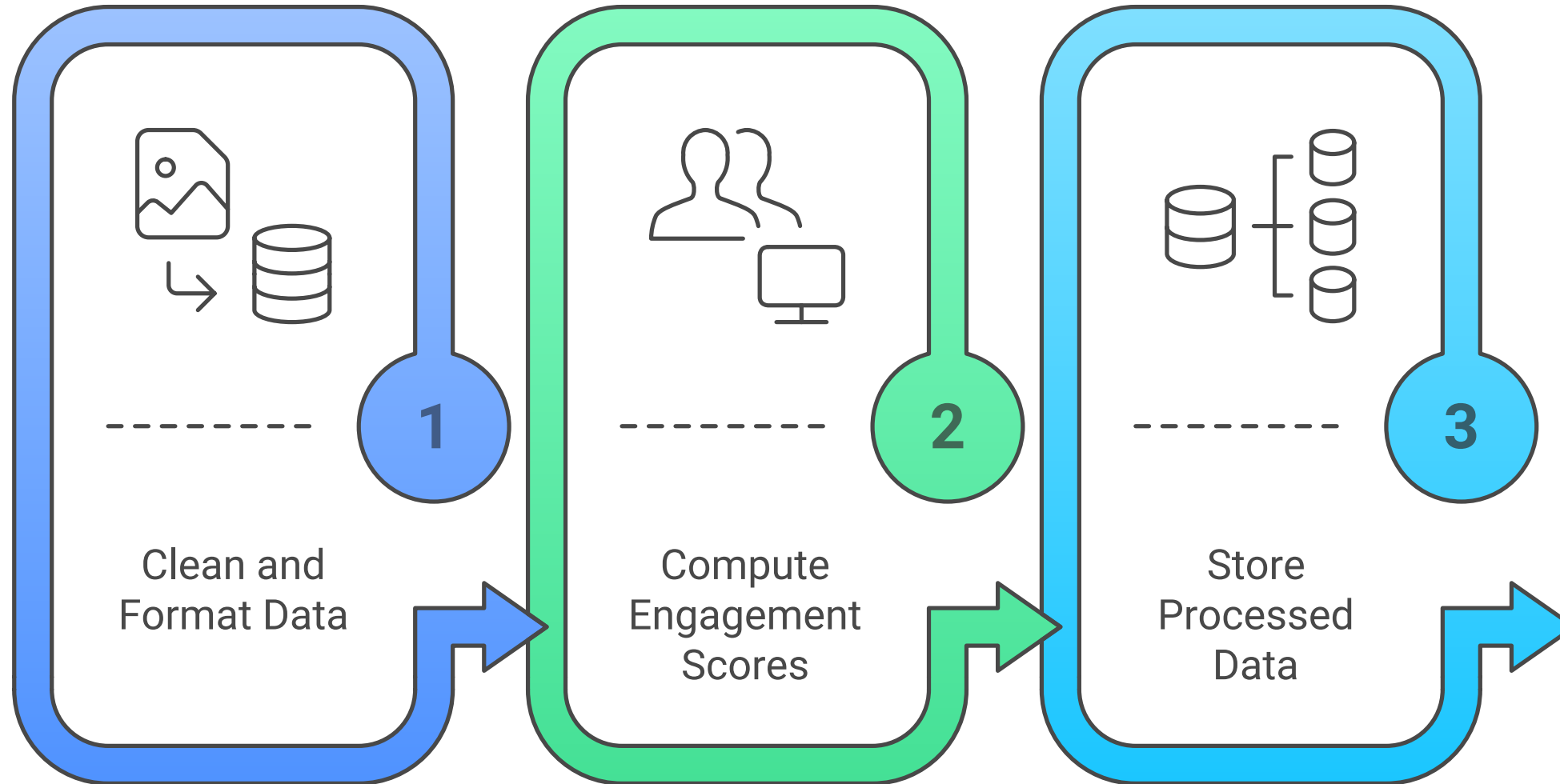
Store Raw Data

Store data in Amazon S3

Step 2: Data Processing & Feature Engineering

- Clean and format data in a **user-item interaction** format.
- Compute **engagement scores** (e.g., watch time, clicks, likes, completions).
- Store processed data in **Redshift / S3** for future use.

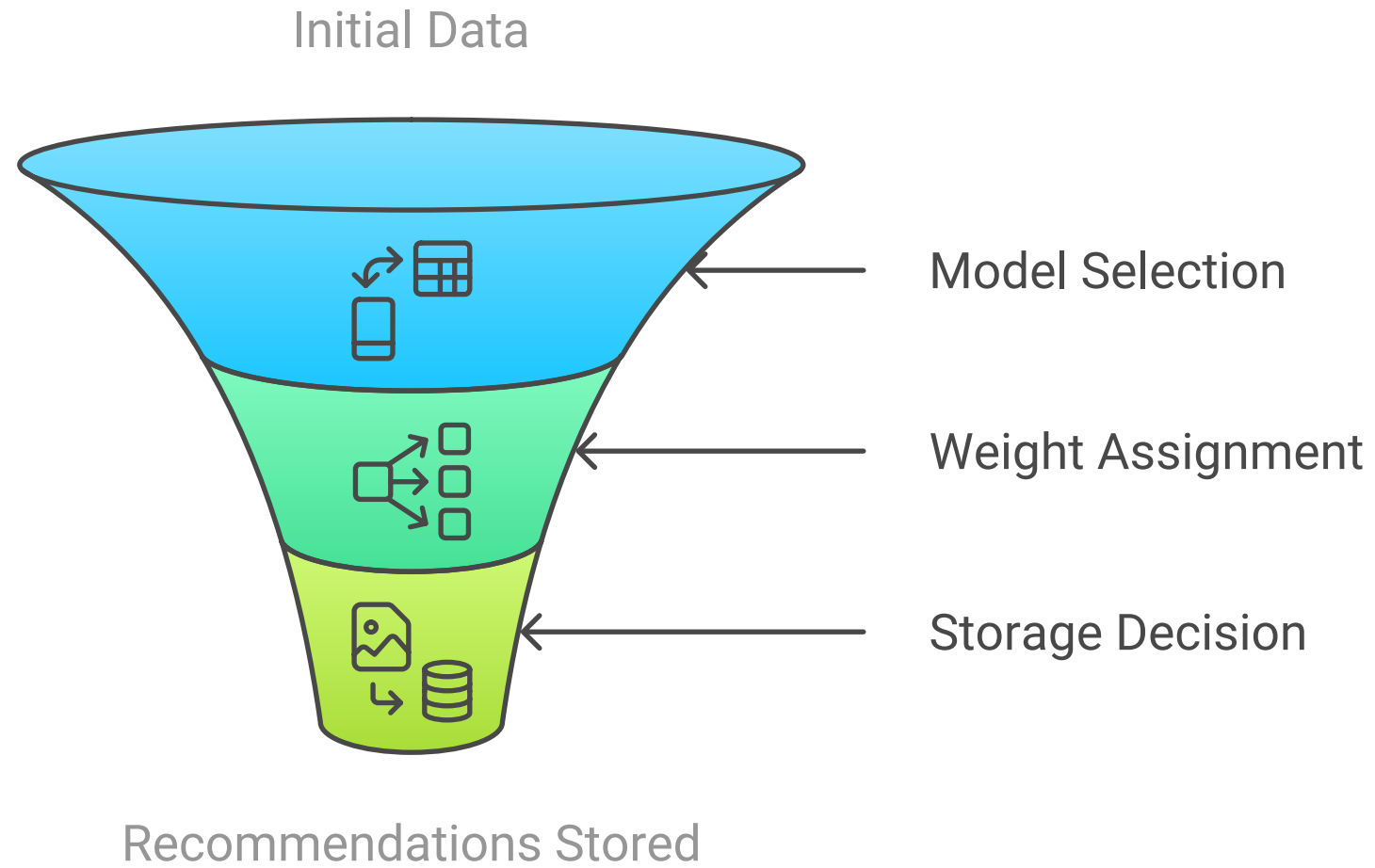
Data Processing and Storage Workflow



Step 3: Generate Baseline Recommendations

- Use a **rule-based** or **collaborative filtering** model.
- Assign weights based on **popularity, user preferences, recency**.
- Store recommendations in **Amazon RDS / DynamoDB**.

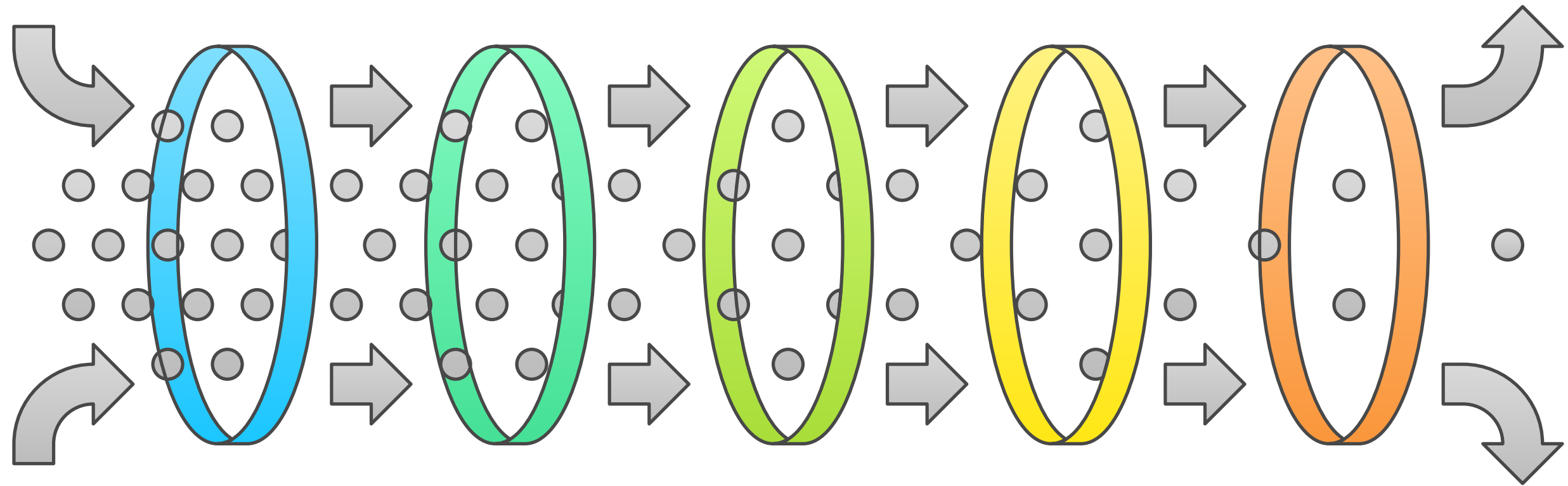
Recommendation Generation Process



Step 4: Train Amazon Personalize Model

- Create datasets in **Amazon Personalize**:
 1. **Users Dataset** – Profile & subscription data.
 2. **Items Dataset** – Content metadata.
 3. **Interactions Dataset** – User activity logs.
- Train the model using **HRNN, SIMS, or Personalized Ranking algorithms**.
- Deploy a **Personalize Campaign** to generate real-time recommendations.

Building a Recommendation System



Create User Dataset

Establishing user profiles and subscription data

Create Items Dataset

Compiling content metadata for items

Create Interactions Dataset

Logging user activities to capture interactions

Train Model

Applying HRNN, SIMS, or Personalized Ranking algorithms

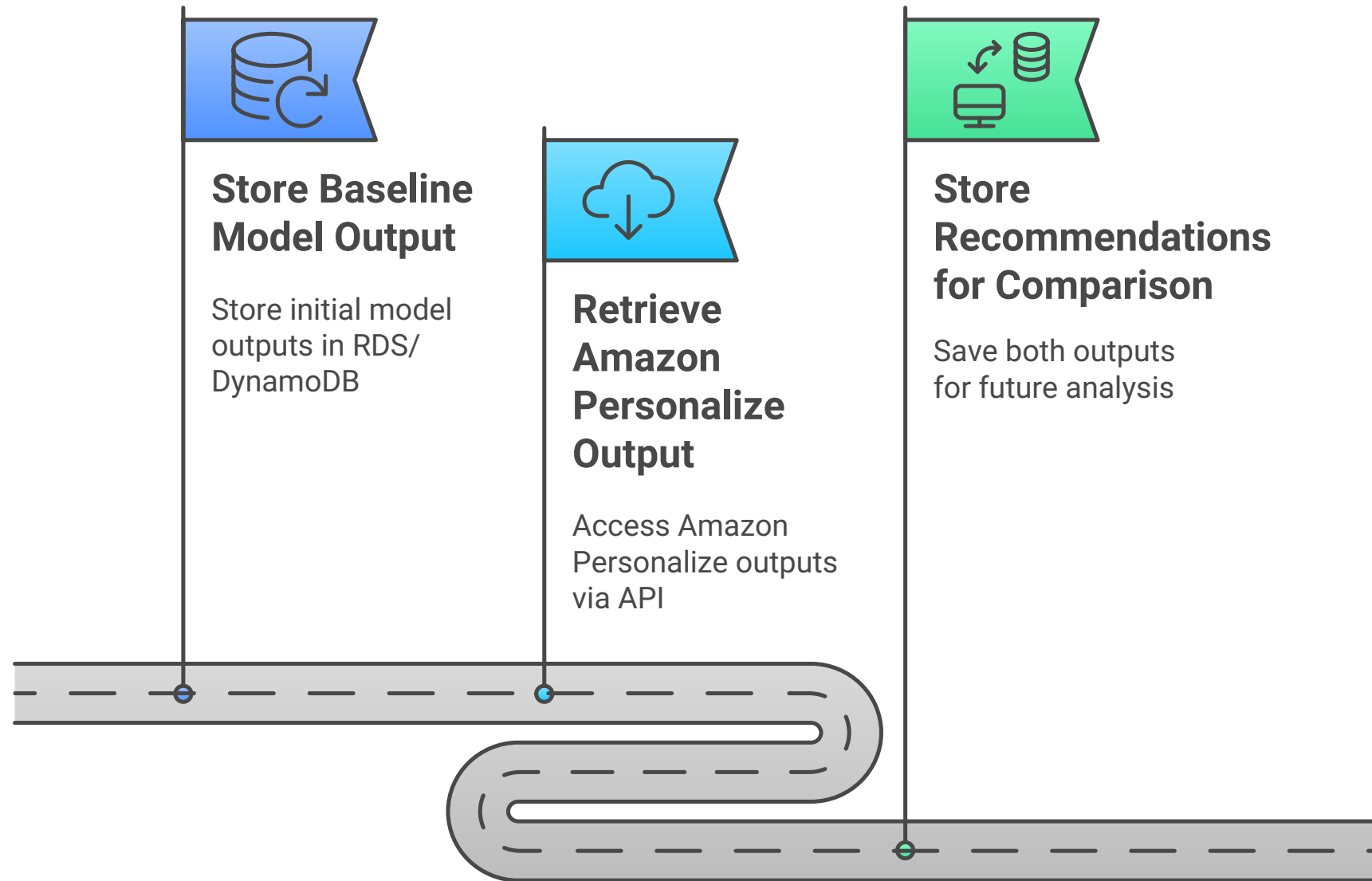
Deploy Campaign

Launching a campaign to generate real-time recommendations

Step 5: Generate & Store Recommendations

- **Baseline Model Output** → Stored in **RDS / DynamoDB**.
- **Amazon Personalize Output** → Retrieved via **Personalize API**.
- Store both recommendations for **comparison**.

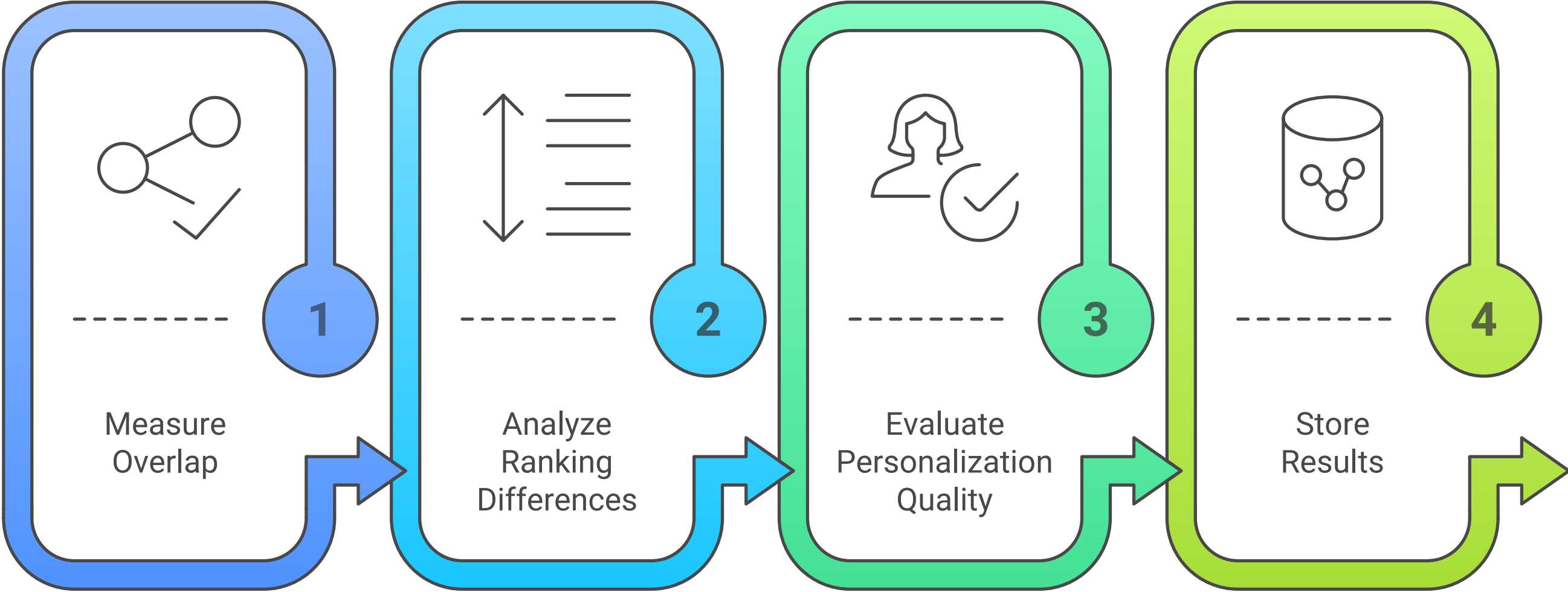
Recommendation Storage and Retrieval Process



Step 6: Compare Recommendations

- **Measure Overlap:** Identify common recommendations.
- **Analyze Ranking Differences:** Compare ranking shifts.
- **Evaluate Personalization Quality:** Check engagement rates.
- Store results in **S3 / Redshift** for visualization.

Evaluation and Storage of Recommendations



Step 7: Generate Reports & Insights

- Use **Amazon QuickSight / Grafana** for dashboards.
- Generate reports on:
 - **Match %** between baseline & Personalize.
 - **New Content Discovery** – Unique items recommended by Personalize.
 - **Engagement Metrics** – Click-through rate, watch time, etc.

Report Generation



Match Percentage

Report on the match percentage between baseline and Personalize.

Report on unique items recommended by Personalize.

New Content



Engagement Metrics

Report on click-through rates and watch time.

Data Flow for Amazon Personalize

