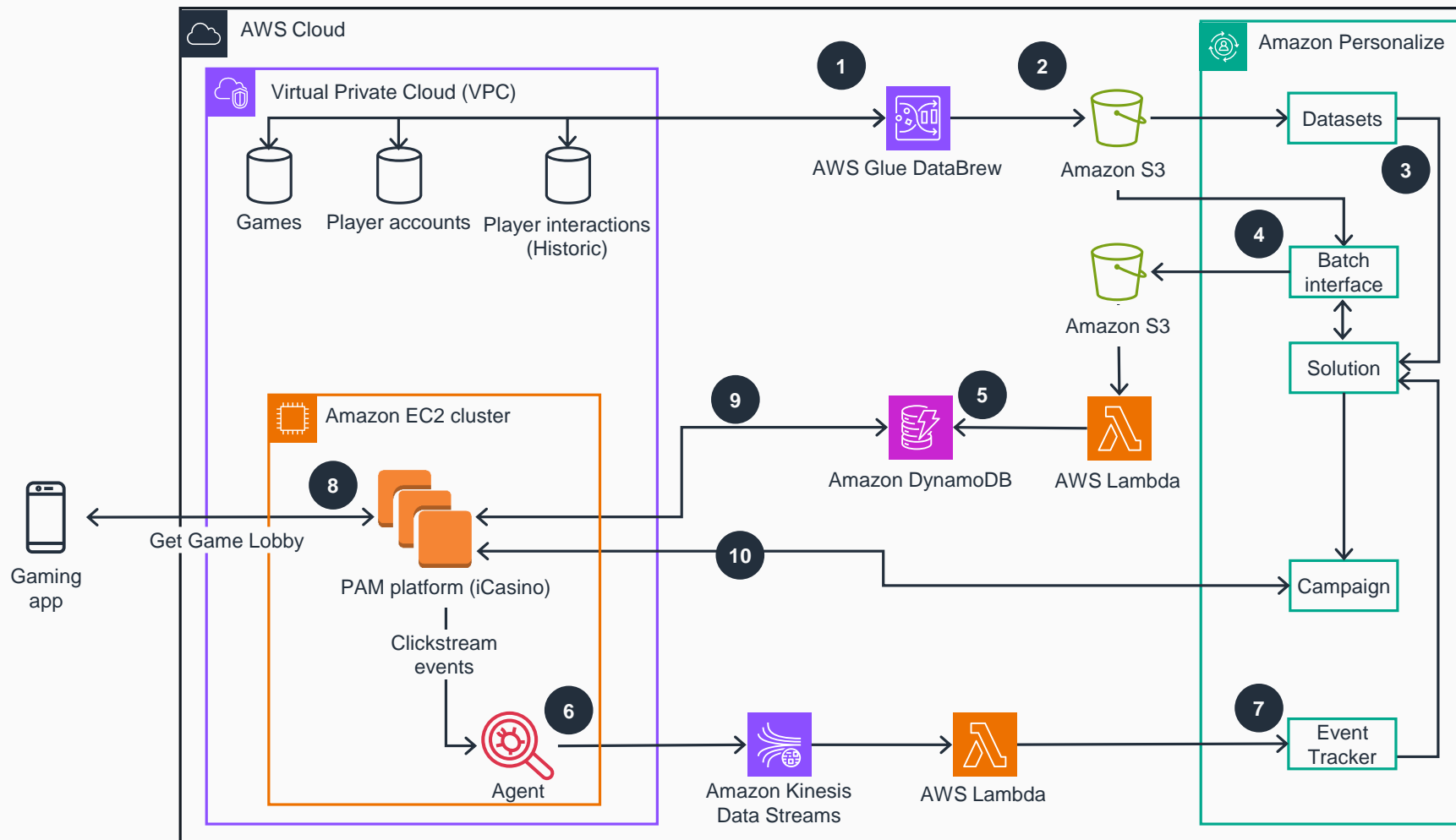


Guidance for Patron Engagement Using Amazon Personalize

This architecture diagram shows how to build a custom, personalized game lobby with recommended games using Amazon Personalize.



- 1** **AWS Glue DataBrew** connects to a database to retrieve games (items), player accounts (users), and player interactions (historic user interaction) data.
- 2** **AWS Glue DataBrew** transforms the data to create .csv training files uploaded to **Amazon Simple Storage Service (Amazon S3)** for upload to **Amazon Personalize** datasets and a games JSON file uploaded to **Amazon S3**.
- 3** **Amazon Personalize** datasets train the models backing custom solution versions, a combination of a recipe (algorithm) and training parameters.
- 4** The **Amazon Personalize** batch interface uses the item-to-item similarities (SIMS) recipe to create a batch file of game recommendations saved both to **Amazon S3** and as custom solution versions.
- 5** **AWS Lambda** retrieves game recommendations from **Amazon S3** and imports them to **Amazon DynamoDB**.
- 6** **Amazon Kinesis** Agent collects real-time player interaction (clickstream) events and sends the events to **Amazon Kinesis Data Streams** with a **Lambda** consumer.
- 7** **Lambda** normalizes player interaction events by transforming them to match the custom solution training parameters and forwarding them to the **Amazon Personalize** event tracker for ongoing custom solution version training.
- 8** The gaming app requests games from the player account management (PAM) platform to populate a "recommended games" section of the lobby user interface (UI).
- 9** The PAM uses the player's last played game to request similar games from **DynamoDB**, used to populate a UI component linking the player's last gameplay to similar games.
- 10** In parallel, the PAM requests real-time, personalized game recommendations from **Amazon Personalize** campaigns and the deployed custom solution version to populate UI components of popular games, trending games, and games tailored for the player.

