The Player Relations Management (PRM) tool puts to use the data collected, processed, and aggregated by Data Platform.

Segments

What are segments?

A **segment** is a mark that is applied to (or removed from) a player account when a certain event is emitted by a Platform service or a Game component. For example, *"first login to the game"* or *"external platform used for first login"* are the segments based on the incoming events from the Authorization service, *"list of payments the player made within the title"*—from the Commerce service, and *"the amount of battles won by a player in the last 30 days"*—from the game itself.

Every segment ever set to a player's account is stored in the player profile and is used to relate this player to a particular group. Thanks to **segmentation** of the player base into player groups, SSE engineers can create personalized player relations (PRM) campaigns with a variety of *triggers*, *condition checks*, and *actions*.

> The created PRM campaigns are also themselves the source of segments. Such technical segments—basically, the marks of where a specific player account ended up as the result of a PRM campaign—are used for statistics, A/B testing (control groups), or creating new campaigns.

A variety of segments are available to games by default after integration with the Platform. These are the basic segments related to Title-specific Platform features (player registration, game sessions, bans, payments, Tracking Engine functionality, etc.) and the player account in Wargaming Universe.

For the cases not covered by basic segments, there is a possibility to create custom game-specific segments based on previously registered data streams via Platform Tools. For instructions on creating segments in Platform Tools, see Platform Tools guide.

How segments are used?

Segments can be used in PRM campaigns to:

- **Trigger** the start of a campaign. When the value and/or timestamp of such segment changes, all matching campaigns are triggered.
- **Check conditions** in campaign formulas. Several segments can be included in one check with AND and/or OR operators in any possible combinations.

  > For example, with one check, you can cover all accounts that *a) last logged in over a year ago* and either *b) demonstrated specific results in the last three battles* or *c) have a history of buying the Premium account*.

- **Take action** on player accounts by **setting, removing, or updating a segment** value.

Kinds of segments

To understand what PRM campaigns can be built on segments, it's useful to know what technical kinds of segments exist in Data Platform.

Basically, a segment stores key-value pairs either as a single record (Single Earliest/Latest kind) or as a list of records (Simple Collection Earliest/Latest and Time-series Collection Earliest/Latest kinds). For a detailed description of segment kinds, see below:

Single segments

Single segments have a timestamp and a value. Segments of this type are something like a label put on a player. For example, if the player has ever made a real purchase, we can label them as a "payer".

Kind	Example	Description
Single Earliest	*The amount of money spent on the first purchase in Premium Shop.*	Only the chronologically earliest information will be retained.
Single Latest	*Fact and date of the latest account login into WG universe.*	The last login value is updated with every new login, and only the last timestamp modification is kept.

This kind of segments may have a limited time-to-live (TTL) and be removed from the player account automatically upon expiration.

Simple collection segments

Simple collection segments act like a key-value inventory (e.g. the list of tanks/battle achievements/number of battles per vehicle/balance per currency, etc.).

In such a segment, objects (e.g. tanks) are differentiated by unique keys. So, if the segment gets new data of the already stored tank, no new records appear. Instead, the value and the timestamp fields of that tank get updated.

Kind	Example	Description
Simple Collection Earliest	*First battle results per each tank.*	The tank is in the key (via glossary), the battle result is in the value. Only initial records of key-value pairs are stored.
Simple Collection Latest	*The total number of battles per each of the four game modes.*	Key is the mode (i.e. regular/clan/epic/ranked), value is the total. Existing records are updated with the latest data that comes with the latest timestamp modification.

No limits on the amount of records/their TTL.

Time-series collection segments

Time-series collection segments store a sequence of records where the timestamp plays the role of the key, and the object (value) is not unique—for example, logging every battle on a selected tank.

This type is useful when it’s important to have chronologically ordered records (e.g. payment transactions, battle results to count a series of wins, etc.) over a specified period of time (for example, 30 days).

Kind	Example	Description
Time Series Collection Earliest	*First 10 redeemed bonus codesets.*	Only the earliest records are kept based on their timestamp (timestamp is the redemption date/time).
Time Series Collection Latest	*A chronological list of the vehicles used in the battles during the last 14 days, ordered by the battle day.*	Only the latest records are kept based on their timestamp (timestamp is when the battle started).

Segments of this type always have limits on the amount of records and/or TTL for the record.

PRM rules

What are PRM rules?

PRM rules set the logic of personalized player relations management (PRM) campaigns.

A **rule** contains the algorithms of triggering checks of accumulated segments for a particular account and subsequently taking action on this account if the check conditions are satisfied. The building blocks of a rule are:

- **Trigger**—the trigger for a campaign launch. Can be continued with condition check(s) and an action.
- **Conditions check**—conditions with AND/OR relations, against which the account will be checked. A condition can be based on the mere fact of a segment presence or contain formulas involving segment values, operators, etc. For an account that is being checked, the check forms two branches of logic—the true branch and the false branch—each of which can lead to an action, be continued with another check, marked for tracking, or left without processing.
- **Action sets**—actions that should be applied to the account if it reaches a particular rule branch in the check tree. Possible actions include: notification sending, product granting, accrual/dismissal of an entitlement, account suspension, segment setting, and more.

> ### Examples of campaigns built with PRM rules
>
> *Email notification AND/OR granting of the item (product/entitlement) to the player who returned to the game after {N} days of inactivity.*
>
> *Sending an email with {content} to the player’s email of registration after the fact of purchase for real money the product with a specific {product id} AND if the additional conditions {the player hasn’t received any other monetization proposals for {number of days} are satisfied.*
>
> Note that the content and design of the emails, as well as the granted entitlements/products must be created/configured beforehand. For more examples of PRM campaigns, see PRM campaigns.