Always on measurement

Created by	Tobi Konitzer
Created time	May 4, 2026 1:57 PM
Last edited time	May 4, 2026 8:59 PM
Type	Technical Spec
Related Docs	https://www.figma.com/board/yyjO8zou809KSa6KPiDmhd/AI-Foundations-CME?node-id=3166-858
Last edited by	Tobi Konitzer
Status	In progress

Summary and Context

One of the key pain-points of our customers is not being able to measure the value of the GrowthLoop-orchestrated experiments over time, once the treatment group is scaled up to 100% of traffic.

> 💡 Note: This is only relevant for audiences that have an *ongoing export*

This will allow customers to understand effects of any intervention/experiment in perpetuity, while also serving as an early warning system for experiments/exports that are scaled up, i.e. it will allow our customers to be alerted to scaled exports or any experiments for that matter that may no longer generate positive ROI.

The Built-In Early Warning System

This serves as an automated alert:

- When has the market shifted?
- When has your strategy lost its edge?
- What scaled decisions should be scaled back?

This project will build always-on incrementality measurement for any experiment in *audiences* for *any audiences that have an ongoing export.* TL;DR:

(1) Always on measurement uses counterfactual ML during the experiment time to predict outcome under treatment and under control

(2) post experiment runtime, when the audience export is scaled to 100%, the UI/UX will show aggregated predicted outcomes by treatment vs. aggregate predicted outcomes by control as lift

Proposed Solution & Dependencies

ML

This requires running counterfactual ML on the outcome on experiment assignment and a handful of features, and using the predicted outcomes under treatment and control to show causal lift. See Appendix 1 — ML approach for details.

Implementation

Currently, the charts for *standard* measurement for experiments of audiences with *one-time export* live under performance.

Europe Customers

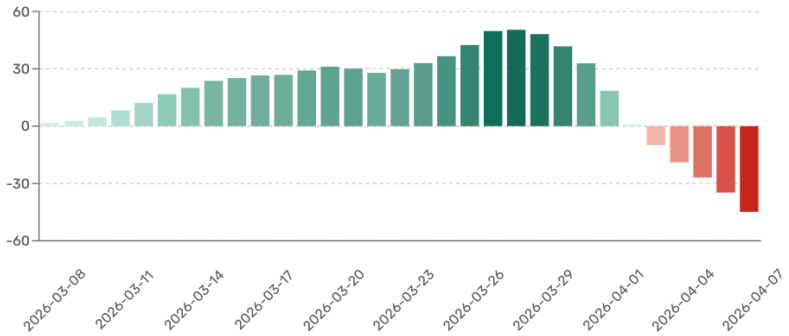
Inactive Export | intelligence xp demo | anthony | Last saved 13 minutes ago | Add tags

Overview | Demographics | Insights | Exports | Performance | Queries | Settings

Intelligence_xp_demo_metric ×

StatSig	Cumulative Impact	% Lift
36% ✓	-90.8K	-1.8%
Target achieved	Cumulative incremental gain	Treatment vs control

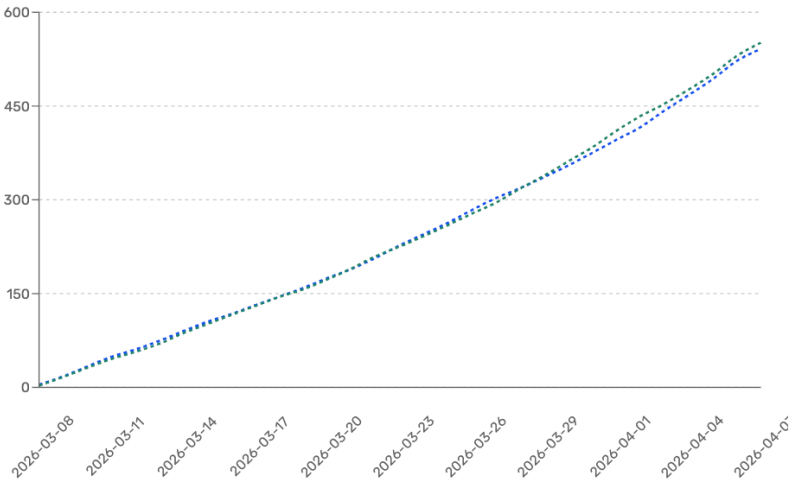
Cumulative Impact



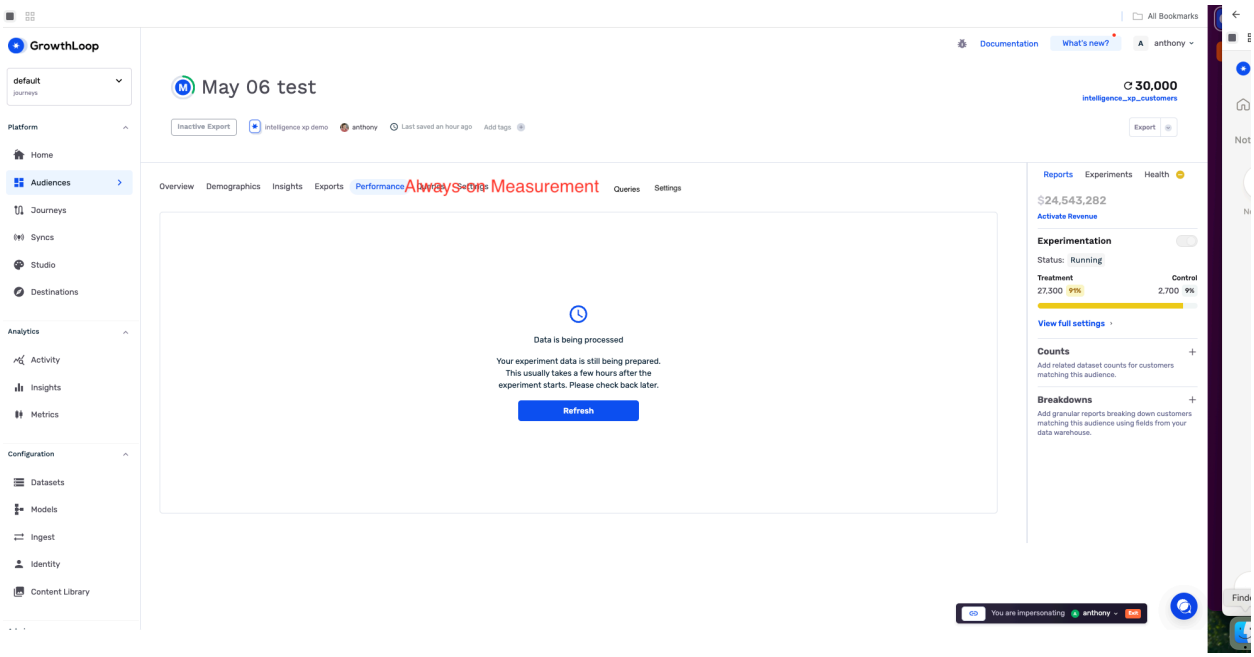
Treatment

541.18

Control Treatment

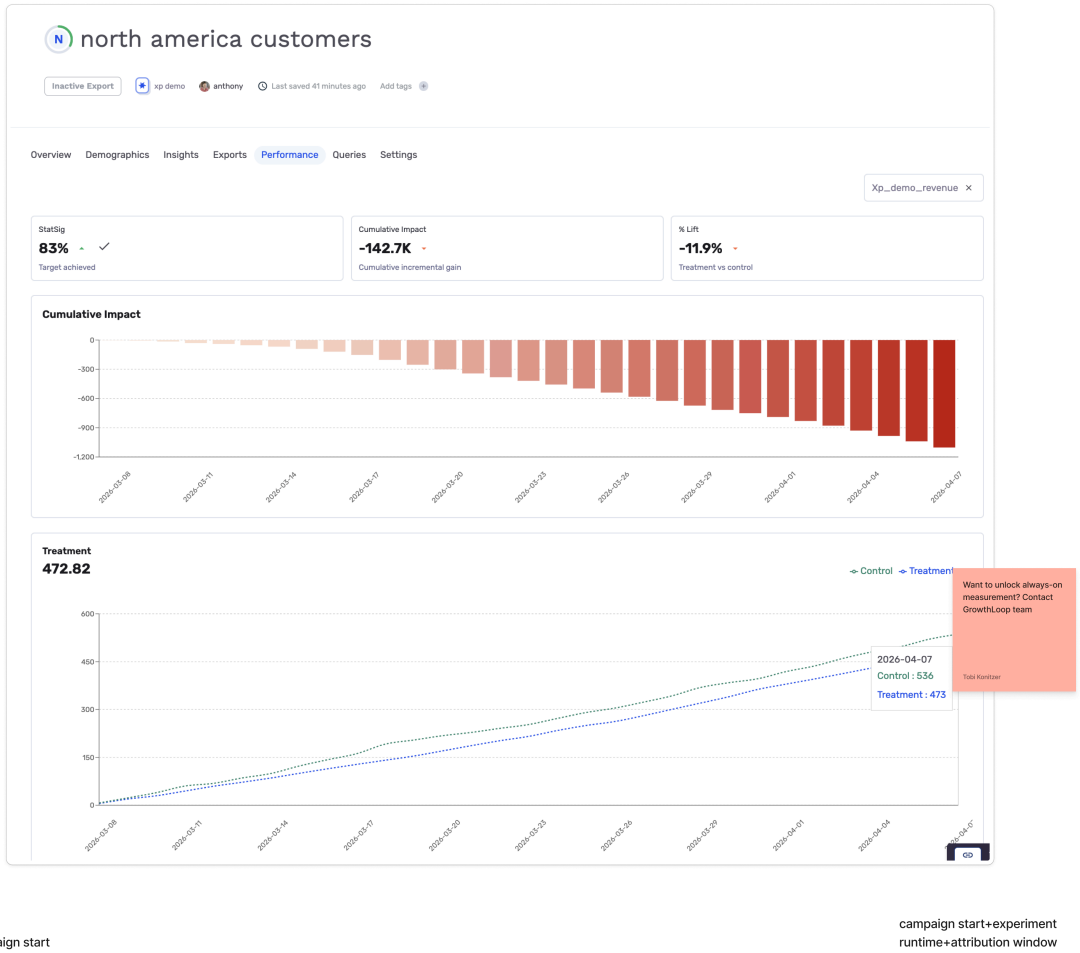


Always-on measurement will use the same visualization schema, but should live on a separate tab.



Flow always-on measurement *current*

1. Customers see the end of the measurement time-frame for any audiences with an ongoing export (below)



campaign start

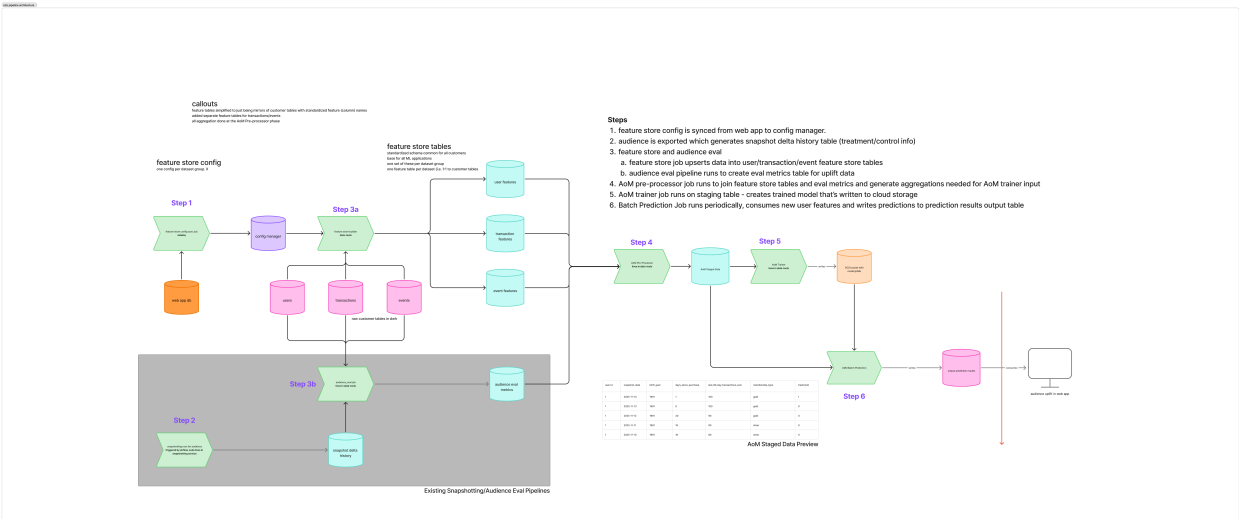
campaign start+experiment runtime+attribution window

2. They see a blurred graph that starts at the end of measurement timeframe, with the node to “activate always-on measurement, contact your Account Manager
3. The Feature Selection part of the ML pipeline is done manually running python on the customer compute using a warehouse-specific work-around we developed,
4. model is run via data tools (using pyhthon compute)
5. Inference job is run and prediction table is created with the current ML pipeline in data-tools
6. Dashboards update

Flow always-on measurement desired

1. same as above
2. same as above
3. Interacts with feature store, automatically selects features
4. Same as above
5. Same as above
6. Same as above

Architecture



Appendix 1 — ML Approach

- c : Cohort of shoppers; c=0 for “control” cohort that do NOT experience audience export, and c=1 for the “treatment” cohort that experience audience export .
- i : An individual user comprised of distinct features *to be determined*
- $\theta_{c,i}$ Continuous KPI/outcome/reward, e,g, real-time customer revenues per user . under treatment (c=1) and control (c=0)

The causal Average Treatment Effect (ATE) is now defined as

$$\text{ATE} = \mathbb{E}_i\left[\theta_{1,i} - \theta_{0,i}\right] = \mathbb{E}_i[\theta_{1,i}] - \mathbb{E}_i[\theta_{0,i}]$$

Note: This average treatment effect is never observed because we don't observe observations under treatment *and* control simultaneously (fundamental problem of causal inference)

Measurement strategy

We can estimate $[\hat{\theta}_{1,i}, \hat{\theta}_{0,i}]$ from a Machine Learning model that generically predicts
$\hat{\theta}_i$:

$$\hat{\theta}_i = \sum_{m=1}^{M} f_m(c_i, x_i), \quad f_m \in \mathcal{F}$$

where $f_m$is a representation of regression trees, specifically XGBoost. Now, $\hat{\theta}_{1,i}$ and $\hat{\theta}_{1,i}$ can be derived by simply setting c to 1 or 0.

Model drift/future productization

- We can use $\text{Median}\left(\left|\hat{\theta}_{c=1,i} - \theta_{c=1,i}\right|\right)$to monitor drift specifically for observations under treatment, for which we have actual and predicted treatments. For now, we don't intend to do anything with it, but we could prdouctize re-experimentation when drift is above a threshold for future builds.