# Recurring Detection Algorithm Methodology

## Background

The Fundraising Effectiveness Project (FEP) pools transaction data from multiple providers of donor management systems (DMS), the software that nonprofit organizations use to record and manage donor relationships and online fundraising platforms. Because the FEP draws from many different providers, the data it receives is not uniform: providers differ in what fields they capture, how they define recurring giving, and how consistently they pass that information through to the FEP database.

Two gaps are particularly relevant here.

First, while all providers now supply a recurring flag for their transactions, this has not been applied retrospectively to all historical data uploads. This means a significant share of the FEP transaction pool, particularly for earlier years, arrives with no information on whether a gift is part of a scheduled series.

Second, even where recurring flags are present, they do not always capture the full picture: when a donor sets up a recurring gift through a third-party fundraising tool (such as a peer-to-peer platform, or an embedded giving widget), that transaction may be processed and recorded in the DMS without the recurring tag being passed along. The result is that provider-supplied recurring flags, even where available, can sometimes undercount recurring activity.

To address both gaps, we developed an algorithm that infers recurring status directly from transaction patterns, without relying on provider-supplied flags. This approach is applied uniformly across the full FEP transaction pool, producing a consistent recurring label regardless of which provider contributed the data.

## Methods

The FEP's recurring detection algorithm identifies recurring donation patterns from raw transaction data, without relying on provider-supplied flags — necessary because such flags are inconsistently supplied across providers and platforms (see Background). The core matching logic is straightforward: for a given donor, organization, and gift amount, the algorithm searches for sequences of transactions that arrive at regular, predictable intervals.

The algorithm detects eight schedule cadences, grouped into two families:

- **Monthly-based** (defined by day-of-month): monthly, bi-monthly, quarterly, semi-annual, annual
- **Weekly-based** (defined by day-of-week): weekly, bi-weekly, four-weekly

This distinction matters because the two families use different matching logic. Weekly-based cadences check whether transactions recur on the same day of the week at the expected number of weeks apart. Monthly-based cadences check whether transactions fall on the same day of the month at the expected number of months apart, with special handling for end-of-month dates and February.

A transaction is flagged as recurring when it belongs to a series of at least **three consecutive transactions** arriving at the expected interval. Each flagged transaction receives a schedule identifier encoding the cadence type and unique schedule to which it belongs, enabling downstream aggregation into schedule-level records.

Real-world recurring donors occasionally miss a single payment, such as when a credit card expires. To handle this, the algorithm permits one skipped interval within an otherwise valid series. The gap is bridged and the missing transaction position is recorded as a skipped payment, while still being labelled as part of the recurring schedule, so that a single lapse does not artificially split one long-running schedule into two shorter ones.

Beyond individual transaction flags, the algorithm produces schedule-level summary records capturing each series' cadence, start and end date, gift amount, and transaction count. Related schedules for the same donor, such as when a donor upgrades their monthly amount mid-year, creating two consecutive series at different amounts, are chained into unified giving episodes. These episodes underpin donor lifecycle annotations that distinguish, for example, a one-off gift made by an active recurring donor from transactions that belong directly to the recurring schedule.

## Validation

We tested the algorithm against a dataset with approximately 4 million natively flagged recurring transactions from a single data provider. The aim was to assess recall (how many provider-flagged recurring transactions the algorithm independently detected) and to quantify the additional transactions it identified beyond the provider's own flags.

The algorithm detected 97.7% of the natively flagged recurring transactions, indicating high recall. Of the 2.3% missed:

- Around 60% were complete misses: the algorithm detected no recurring pattern for that donor-org-amount combination at all. In nearly all of these cases, the series contained fewer than three transactions and therefore did not meet the minimum threshold for detection.

- The remaining 40% were partial misses: the algorithm did identify some recurring transactions for the same donor-org-amount combination, but the specific missed transactions fell outside a detected run, typically because of an irregular gap in the schedule that broke the sequence.

Beyond the provider-flagged transactions, the algorithm identified an additional 4.7 million transactions as recurring, more than double the natively flagged set. This suggests that provider flags are significantly undercounting recurring activity, likely because donations initiated through third-party fundraising tools are not always passed back to the donor management system with a recurring tag. The algorithm's higher count is therefore expected and desirable, not a sign of over-detection, though the limitations section below notes that false positives are possible.

This validation was conducted against a single provider's flagging system. Because provider flagging practices vary, the algorithm's performance may differ across other data sources in the FEP pool. We aim to test the algorithm against more providers in future.

## Limitations

**Minimum threshold requirement.** The algorithm requires at least three consecutive transactions at the correct interval to flag a series as recurring. This means recently established schedules — particularly those with lower-frequency cadences like quarterly or annual — may not accumulate enough transactions to be detected within the observation window. As a result, recurring activity is likely understated for recent periods and for less frequent schedule types. This has downstream consequences for calculations of metrics such as donor retention, which will be inflated for recurring donors since it misses donors churning before the threshold.

**Potential false positives.** The algorithm may flag non-recurring transactions that happen to share the same amount, organization, and timing pattern. Candidates include pledge installments, membership dues, or payroll deduction programs. These would inflate recurring counts without representing a true donor-initiated recurring gift. We do not have a reliable way to estimate the false positive rate from available data at this stage.

**Irregular but repeating schedules are not captured.** The algorithm applies strict interval rules, so donors who give the same amount to the same organization on a recurring but irregular basis — for example, every year but not on the same calendar date — will not be detected. This is a deliberate design choice that keeps precision high but means some recurring donors are categorized as one-time or lapsed.

**Upgrades and downgrades may be missed.** Because the algorithm anchors on a fixed gift amount, a donor who increases or decreases their recurring gift amount will appear

to have ended one schedule and started another. This is partially handled by the chained episode logic, but the transition period may result in some recurring transactions being missed or misattributed, at least until a donor meets the three consecutive transaction threshold in their new schedule.

## Looking ahead

This methodology represents a first iteration of our work in this space. Our aim is to be transparent in how recurring giving is identified in our data, and to invite feedback from practitioners, researchers, and data providers who work in this space.

We expect to refine the algorithm over time in a number of directions. One near-term priority is incorporating additional transaction characteristics, particularly the online/offline split and tender type (such as credit card, ACH, check, or payroll deduction), into both the detection logic and the reporting. These dimensions likely correlate with recurring giving patterns in meaningful ways: payroll deduction programs, for instance, may explain much of the biweekly schedule behavior observed in this report, and online versus offline channels may differ substantially in how recurring relationships are initiated and maintained. Bringing these distinctions into the methodology would allow for sharper analysis and more actionable findings.

Further out, we hope to explore the development of sector-wide benchmarks for recurring giving, standardized metrics that organizations can use to assess their own recurring programs against peers of similar size, cause area, or donor profile.