Table of Contents

# UV Aggregation Strategy for Multi-Level Reporting

## Executive Summary

This document proposes a solution for calculating Unique Visitor (UV) metrics across different time periods (7, 14, 21 days post-publishing) at aggregated levels (Site, Division, Region) while maintaining consistency with article-level reporting.

**Key Decision Required:** How should we count unique visitors when aggregating multiple articles with different publishing dates?

## 1. Problem Statement

### Current State

* We successfully calculate UV metrics for **individual articles** based on fixed time periods after their publishing date
* Example: Article A published on Jan 1 → “First 7 Days” = Jan 1-7

### The Challenge

When users apply filters at aggregated levels (Site, Division, Region), we encounter a fundamental question:

**How do we calculate “First 7 Days” when a Site contains 100 articles published on different dates?**

### Real-World Example

Consider the “News & Events” site with these articles:

| Article | Publishing Date | First 7 Days Period |
| --- | --- | --- |
| Article A | Jan 1, 2023 | Jan 1 - Jan 7 |
| Article B | Feb 15, 2023 | Feb 15 - Feb 21 |
| Article C | Mar 20, 2023 | Mar 20 - Mar 26 |

**Question:** When a stakeholder filters by “News & Events” and views “First 7 Days” metrics, what should they see?

## 2. Proposed Solutions

### Option 1: Total Interactions (Duplicates Allowed)

**Concept:** Sum up the individual article metrics. If the same visitor viewed multiple articles in their respective launch periods, count them multiple times.

**Business Question Answered:** *“How many total UV interactions did our articles receive during their launch phases?”*

#### Example Calculation

| Article | Publishing Date | Visitors in First 7 Days | Visitor IDs |
| --- | --- | --- | --- |
| Article A | Jan 1 | 3 visitors | V1, V2, V3 |
| Article B | Feb 15 | 3 visitors | V2, V4, V5 |
| Article C | Mar 20 | 3 visitors | V3, V6, V7 |
| **Site Total** | - | **9 interactions** | (duplicates allowed) |

**Interpretation:** “Articles in News & Events generated 9 UV interactions across their first 7 days post-publishing.”

#### Advantages

* Simple to calculate and understand
* Reflects total engagement volume
* Directly comparable to article-level metrics (just a sum)
* Useful for measuring content production effectiveness

#### Challenges

* Overstates unique audience reach
* Same visitor counted multiple times
* Can be misleading if stakeholders expect unique counts
* Higher numbers might create false performance expectations

### Option 2: Unique Reach (Deduplicated)

**Concept:** Count each visitor only once, even if they visited multiple articles during their respective launch periods.

**Business Question Answered:** *“How many unique people did we reach with our content during article launch periods?”*

#### Example Calculation

| Article | Publishing Date | Visitors in First 7 Days | Visitor IDs |
| --- | --- | --- | --- |
| Article A | Jan 1 | 3 visitors | V1, V2, V3 |
| Article B | Feb 15 | 3 visitors | V2, V4, V5 |
| Article C | Mar 20 | 3 visitors | V3, V6, V7 |
| **Site Total** | - | **7 unique visitors** | V1, V2, V3, V4, V5, V6, V7 |

**Interpretation:** “News & Events reached 7 unique visitors across all articles’ first 7 days post-publishing.”

#### Advantages

* True measure of unique audience reach
* No double-counting of visitors
* Better reflects actual market penetration
* Aligns with marketing concepts of “reach” vs “frequency”

#### Challenges

* More complex to calculate
* Time periods overlap (Jan-Mar in example)
* Less intuitive: “First 7 Days” doesn’t mean a continuous 7-day period
* Cannot simply sum article-level metrics

### Option 3: Hybrid Approach (Recommended)

**Concept:** Provide BOTH metrics in the dashboard to answer different business questions.

#### Dashboard Example

**Site: News & Events**

| Metric | First 7 Days | First 14 Days | First 21 Days |
| --- | --- | --- | --- |
| Unique Reach | 1,250 visitors | 2,340 visitors | 3,120 visitors |
| Total Interactions | 1,580 interactions | 2,890 interactions | 3,850 interactions |
| Overlap Rate | 26% | 23% | 23% |
| Articles Included | 25 articles | 25 articles | 25 articles |

**Timeframe Coverage:** Jan 2023 - Dec 2023

#### Metric Definitions

| Metric | Definition | Use Case |
| --- | --- | --- |
| **Unique Reach** | Distinct visitors who viewed at least one article during its first X days | Measure audience size and market penetration |
| **Total Interactions** | Sum of all article-level UV counts (duplicates allowed) | Measure content engagement volume |
| **Overlap Rate** | Percentage of repeat visitors across articles | Understand audience loyalty |

#### Advantages

* Answers multiple business questions
* Transparent methodology
* Stakeholders can choose relevant metric
* Provides context through overlap rate

#### Challenges

* Requires more dashboard space
* Need to educate stakeholders on difference
* Slightly more complex implementation

## 3. Filter Scenarios

### Scenario 1: Site Filter Only

**Filter:** Site = “News & Events”

**Result:** All visitors who viewed any article in “News & Events” during that article’s first X days post-publishing.

**Example:** - Site contains 25 articles published between Jan-Dec 2023 - Unique Reach (First 7 Days) = 1,250 visitors - These 1,250 people visited at least one article within 7 days of that article’s publication

### Scenario 2: Division Filter Only

**Filter:** Division = “Marketing”

**Result:** All visitors from the Marketing division who viewed any article during its first X days post-publishing.

**Example:** - Marketing division visitors viewed articles across multiple sites - Unique Reach (First 7 Days) = 450 visitors - These 450 Marketing users visited at least one article within 7 days of publication

### Scenario 3: Combined Filters

**Filter:** Site = “News & Events” AND Division = “Marketing”

**Result:** Marketing division visitors who viewed News & Events articles during their first X days post-publishing.

**Example:** - Intersection of Site and Division filters - Unique Reach (First 7 Days) = 180 visitors - These 180 Marketing users visited News & Events articles within 7 days of publication

## 4. Visual Timeline Example

To illustrate the deduplication concept:

**Site: News & Events - Timeline View**

| Article | Publishing Date | First 7 Days Window | Visitors |
| --- | --- | --- | --- |
| Article A | Jan 1, 2023 | Jan 1 - Jan 7 | V1, V2, V3 |
| Article B | Feb 15, 2023 | Feb 15 - Feb 21 | V2, V4, V5 |
| Article C | Mar 20, 2023 | Mar 20 - Mar 26 | V3, V6, V7 |

**AGGREGATED METRICS FOR SITE**

**Option 1 - Total Interactions:** - Calculation: 3 + 3 + 3 = **9 interactions**

**Option 2 - Unique Reach:** - Union of visitors: {V1,V2,V3} ∪ {V2,V4,V5} ∪ {V3,V6,V7} - Result: {V1, V2, V3, V4, V5, V6, V7} - Total: **7 unique visitors**

**Note:** - V2 visited both Article A and B (counted once in Unique Reach) - V3 visited both Article A and C (counted once in Unique Reach)

## 5. Why Employee Count Matters (130K Internal Users)

### The Internal Intranet Context

Unlike external websites with unlimited potential audience, our intranet has a **fixed maximum audience of 130,000 employees**. This creates unique characteristics:

**Key Implications:**

| Aspect | Internal Intranet (130K employees) | External Website (unlimited) |
| --- | --- | --- |
| **Maximum Reach** | Capped at 130,000 people | Unlimited potential |
| **Visitor Overlap** | High - engaged employees read multiple articles | Lower - diverse audience |
| **Deduplication Impact** | **Critical** - Same employees appear repeatedly | Less critical |
| **Business Question** | “What % of our employees engaged with this content?” | “How many people did we reach?” |

### Real-World Example with Employee Base

**Scenario:** Site “News & Events” with 25 articles published over 3 months

| Metric | Without Deduplication | With Deduplication | Difference |
| --- | --- | --- | --- |
| Total Interactions | 15,000 visits | 15,000 visits | - |
| Unique Visitors | **15,000 “visitors”** | **8,500 employees** | 43% overcount |
| **As % of Company** | **11.5%** (misleading) | **6.5%** (accurate) | 5 percentage points off! |

**Insight:** Without deduplication, we’d incorrectly report that 11.5% of the company engaged, when it’s actually 6.5%.

### Why This Matters for Stakeholders

**Question stakeholders really want answered:**

*“What percentage of our 130,000 employees engaged with our content during the launch period?”*

**With Total Interactions (no deduplication):** - Answer: “15,000 interactions” - Stakeholder thinks: “11.5% of employees engaged” - **Wrong:** Some employees counted 2-3 times

**With Unique Reach (deduplicated):** - Answer: “8,500 unique employees” - Stakeholder thinks: “6.5% of employees engaged” - **Correct:** Each employee counted once

### Employee Engagement Patterns

Typical patterns we see in internal intranets:

| Employee Segment | Behavior | Impact on Metrics |
| --- | --- | --- |
| **Super Users (5%)** | Read 10+ articles per month | Would be counted 10+ times without deduplication |
| **Regular Readers (20%)** | Read 3-5 articles per month | Would be counted 3-5 times |
| **Occasional Readers (40%)** | Read 1-2 articles per month | Minimal duplication |
| **Non-Readers (35%)** | Rarely visit intranet | Not in metrics |

**Without deduplication:** Super users and regular readers inflate the numbers significantly.

**With deduplication:** Get true measure of employee reach.

### Division/Region Filtering Becomes More Meaningful

With a known employee base, division/region filters tell a more complete story:

**Example: Marketing Division (5,000 employees)**

| Filter | Unique Reach | Total Interactions | Interpretation |
| --- | --- | --- | --- |
| Site: News & Events | 1,250 employees | 2,100 visits | 25% of Marketing division engaged, averaging 1.7 articles each |
| Division: Marketing | 3,200 employees | 8,500 visits | 64% of Marketing engaged across all sites |

**Business Value:** - “64% of Marketing division engaged with intranet content in launch periods” - Clear penetration metrics per division - Can compare: “Marketing 64% vs Sales 45%” - which division is more engaged?

### Maximum Possible Reach

With 130K employees, you can calculate theoretical maximum and actual performance:

| Site | Articles | Total Interactions | Unique Reach | % of Company | Avg Articles per Reader |
| --- | --- | --- | --- | --- | --- |
| News & Events | 25 | 15,000 | 8,500 | 6.5% | 1.8 |
| HR Updates | 50 | 45,000 | 22,000 | 16.9% | 2.0 |
| Tech Blog | 30 | 8,000 | 3,200 | 2.5% | 2.5 |

**Insights:** - HR Updates reaches most employees (16.9%) - Tech Blog has lower reach but higher engagement (2.5 articles per reader) - Can set targets: “Goal: Reach 20% of employees in first 7 days”

## 6. Recommendation

**Implement Option 3: Hybrid Approach**

### Rationale

1. **Different Stakeholder Needs:** Marketing teams care about reach, while content teams care about engagement volume
2. **Transparency:** Showing both metrics prevents misinterpretation
3. **Flexibility:** Users can choose the metric that answers their specific question
4. **Best Practice:** Industry standard to report both reach and frequency metrics

### Implementation Priority

**Phase 1:** Implement Unique Reach (deduplicated) as primary metric

**Phase 2:** Add Total Interactions as secondary metric

**Phase 3:** Add calculated fields (overlap rate, per-article averages)

## 6. Dashboard Mockup

### Recommended Display Format

**Filter Selection:** - Site: News & Events - Division: All - Region: All

**Performance Metrics**

| Time Period | Unique Employees | % of Company | Total Interactions | Articles | Avg UV per Article |
| --- | --- | --- | --- | --- | --- |
| First 7 Days | 1,250 | 0.96% | 1,580 | 25 | 63 |
| First 14 Days | 2,340 | 1.80% | 2,890 | 25 | 116 |
| First 21 Days | 3,120 | 2.40% | 3,850 | 25 | 154 |

**Key Insights:** - **Employee Reach:** 2.4% of company (3,120 out of 130,000 employees) engaged with News & Events in first 21 days - **Overlap Rate:** 26% of interactions are from employees who read multiple articles - **Coverage Period:** Jan 2023 - Dec 2023 - **Most Recent Article:** Dec 15, 2023

**Note:** Unique Employees are deduplicated across all articles. Each employee is counted once even if they viewed multiple articles during their respective launch periods.

## 7. How We Calculate Unique Reach: A Step-by-Step Visual Guide

This section explains the process in simple, non-technical terms with visual examples.

### The Big Picture: A 3-Layer Approach

Think of calculating Unique Reach like organizing a large event where multiple speakers present on different days, and you want to know how many unique people attended across all sessions.

#### OUR 3-LAYER DATA SYSTEM

**LAYER 1: Raw Event Log** *(Like an attendance sheet)*

Every time someone visits an article, we record it: - “John visited Article A on Jan 3” - “Sarah visited Article B on Feb 16” - “John visited Article B on Feb 17” ← Same person, different article!

⬇️ **Process Daily** ⬇️

**LAYER 2: Visitor Summary Table** *(Like a master registry)*

For each person, track which articles they visited during launch: - “John: ✓ Visited Article A (first 7 days), ✓ Visited Article B (first 7 days)” - “Sarah: ✓ Visited Article B (first 7 days)”

⬇️ **Query Real-Time** ⬇️

**LAYER 3: Dashboard Metrics** *(What stakeholders see)*

Site “News & Events” - Unique Reach: 2 people (John + Sarah)

*Even though there were 3 total visits, only 2 unique people*

### Step-by-Step Process with Real Example

Let’s walk through exactly how we calculate “Unique Reach” for a Site with 3 articles.

#### **STEP 1: Define Each Article’s Launch Window**

Each article has its own 7-day launch window based on when it was published.

| Timeline | Article A | Article B | Article C |
| --- | --- | --- | --- |
| **January 2023** | **Published: Jan 1**Launch Window (7 Days):Jan 1 - Jan 7 |  |  |
| **February 2023** |  | **Published: Feb 15**Launch Window (7 Days):Feb 15 - Feb 21 |  |
| **March 2023** |  |  | **Published: Mar 20**Launch Window (7 Days):Mar 20 - Mar 26 |

**Key Point:** Each article has its own independent launch window. These windows do NOT need to overlap or be continuous.

#### **STEP 2: Track Who Visited During Launch Windows**

For each visitor, we mark which articles they visited during that article’s launch period.

**VISITOR TRACKING BOARD**

| Visitor | Article A (Launch: Jan 1-7) | Article B (Launch: Feb 15-21) | Article C (Launch: Mar 20-26) |
| --- | --- | --- | --- |
| **John** | ✓ Visited on Jan 3✓ COUNT | ✓ Visited on Feb 17✓ COUNT | ✗ No visit✗ Don’t count |
| **Sarah** | ✗ No visit✗ Don’t count | ✓ Visited on Feb 16✓ COUNT | ✓ Visited on Mar 22✓ COUNT |
| **Michael** | ✓ Visited on Jan 2✓ COUNT | ✗ No visit✗ Don’t count | ✗ No visit✗ Don’t count |
| **Emma** | ✗ Visited on Jan 10✗ **Too late!** | ✗ No visit✗ Don’t count | ✓ Visited on Mar 21✓ COUNT |

**Important Note:** Emma visited Article A on Jan 10, which is AFTER the 7-day launch window, so it doesn’t count.

#### **STEP 3: Count Each Person Only Once**

Now we determine: Did each person visit AT LEAST ONE article during its launch? If yes, count them ONCE.

**DEDUPLICATION: COUNT EACH PERSON ONCE**

| Visitor | Visited Any Article in Launch? | Include in Count? |
| --- | --- | --- |
| John | YES (A + B) | ✓ Count Once |
| Sarah | YES (B + C) | ✓ Count Once |
| Michael | YES (A) | ✓ Count Once |
| Emma | YES (C) | ✓ Count Once |

**RESULT: 4 Unique Visitors**

*Even though John visited 2 articles and Sarah visited 2 articles, we count each person only ONCE for the Site total.*

#### **STEP 4: Compare to Total Interactions**

Here’s how the two metrics differ:

**METRIC COMPARISON: UNIQUE REACH VS INTERACTIONS**

**Individual Article Counts:**

| Article | First 7 Days Visitors | Visitor Names |
| --- | --- | --- |
| Article A | 2 visitors | John, Michael |
| Article B | 2 visitors | John, Sarah |
| Article C | 2 visitors | Sarah, Emma |

**Aggregated to Site Level:**

**Option 1 - Total Interactions (Simple Sum):** - Calculation: 2 + 2 + 2 = **6 interactions** - Interpretation: “6 article visits happened during launch windows” - Use Case: Measure total engagement volume

**Option 2 - Unique Reach (Deduplicated):** - Calculation: {John, Michael} ∪ {John, Sarah} ∪ {Sarah, Emma} = {John, Michael, Sarah, Emma} - Result: **4 unique people** - Interpretation: “4 different people visited at least one article” - Use Case: Measure actual audience size

**Overlap Analysis:** - Overlap: 6 interactions - 4 unique people = 2 repeat interactions - Overlap Rate: 2/6 = **33%** - Interpretation: “33% of visits were from people who visited multiple articles during their launches”

### Visual: Data Flow Architecture

This diagram shows how data flows through our system at scale (774 sites, 40,000 articles).

**DATA FLOW PIPELINE**

| Stage | Raw Data (Continuous) | Processing (Daily) | Dashboard (Real-Time) |
| --- | --- | --- | --- |
| **System** | Website Visits | Nightly Batch Job | Business Dashboard |
| **Volume** | • 50-100M records• Real-time streaming | • Groups by visitor• Flags launch visits | • < 1 sec response• Filters: Site, Division, Region |
| **Frequency** | Every page view | Once daily | When user clicks |
| **Storage** | 20-50 GB (Keep 2 years) | 2-3 GB (Keep 3 years) | Pre-calculated Tables < 10 MB |

**Data Flow:** Raw Data → (Process) → Processing → (Prepare) → Dashboard

**Why This Design?**

1. **Layer 1 (Raw Data):** Too big to query directly - would take 30-60 seconds
2. **Layer 2 (Nightly Processing):** Pre-calculate heavy work overnight (5-10 min)
3. **Layer 3 (Dashboard):** Instant results because hard work already done

**Analogy:** Like a restaurant - Raw ingredients (Layer 1) → Prep work overnight (Layer 2) → Fast service (Layer 3)

### Performance: What Stakeholders Can Expect

**DASHBOARD RESPONSE TIME EXPECTATIONS**

| Filter Selection | Example | Response Time | Experience |
| --- | --- | --- | --- |
| Single Site | “News & Events” | < 1 second | ⚡ Instant |
| Multiple Sites (2-5) | “News & Events + Sports” | 1-2 seconds | ✓ Fast |
| Site + Division | “News & Events” + “Marketing” | < 1 second | ⚡ Instant |
| Site + Division + Region | Complex filter combo | 1-2 seconds | ✓ Fast |
| All Sites (774) | “Company-wide view” | 1-2 seconds | ✓ Fast |
| Ad-hoc complex filters | Multiple sites + divisions | 5-15 seconds | ⌛ Tolerable |

**Query Distribution:** - ✓ 80% of queries: Under 2 seconds (pre-calculated) - ✓ 15% of queries: 5-15 seconds (calculated on-demand) - ⚠ 5% of queries: Up to 30 seconds (very complex filters)

### Scale Context: Our Numbers

To give perspective on the complexity we’re managing:

**SYSTEM SCALE OVERVIEW**

| Metric Category | Value |
| --- | --- |
| **Content** |  |
| Sites | 774 sites |
| Articles | 40,000 articles |
| Avg Articles per Site | ~52 articles |
| **Users** |  |
| Total Employee Population | 130,000 employees |
| Unique Visitors (estimated) | ~100,000 active employees |
| Total Visits (estimated) | ~50-100 million visit records |

**DAILY PROCESSING**

| Metric | Value |
| --- | --- |
| Data Volume Processed | ~500 MB - 2 GB of new visits daily |
| Processing Time | 5-10 minutes (overnight) |
| Storage Required | ~2-3 GB (optimized tables) |

**DASHBOARD QUERIES**

| Query Type | Response Time | Usage % |
| --- | --- | --- |
| Common queries | < 2 seconds | 80% |
| Complex queries | 5-15 seconds | 15% |
| Very complex queries | 15-30 seconds | 5% |

**Key Insight:** By processing data overnight, we turn a 30-60 second calculation into a sub-second dashboard response.

**Important Context - Internal Intranet:** - This is an internal employee intranet with 130,000 employees - Maximum possible unique visitors per article = 130,000 (entire company) - Typical unique visitors per article in first 7 days = 100-5,000 employees - Deduplication is **critical** because engaged employees read multiple articles

### The “Pre-Calculate and Store” Strategy

Think of this like a restaurant preparing ingredients:

**COMPARISON: WITH VS WITHOUT PRE-CALCULATION**

| Approach | WITHOUT Pre-Calculation (Too Slow) | WITH Pre-Calculation (Our Solution) |
| --- | --- | --- |
| **Step 1** | User clicks dashboard | User clicks dashboard |
| **Step 2** | System reads 50-100M records | System reads 774 summary rows |
| **Step 3** | Calculates everything from scratch | Instant result from pre-calculated table |
| **Result** | ⏰ Wait 30-60 seconds | ⚡ Result in < 1 second |
| **Analogy** | “I’ll buy the ingredients, prepare them, cook them, and serve you” → ⏰ 2 hour wait | “Here’s your meal - we prepped everything this morning” → ⚡ 5 minute wait |

**Our Strategy:** 1. **Every night:** Process all raw data → Create summary tables 2. **During day:** Users query summary tables → Instant results 3. **Trade-off:** Data refreshes daily (not real-time), but queries are fast

### Success Metrics: How We’ll Know It’s Working

**SUCCESS CRITERIA CHECKLIST**

| Category | Success Criteria | Target |
| --- | --- | --- |
| **PERFORMANCE** |  |  |
|  | Dashboard query response time | 80% of queries under 2 seconds |
|  | Complex query completion | Under 15 seconds |
|  | Daily data refresh | Completes within 30-minute window |
|  | Concurrent users | System handles 100+ users without slowdown |
| **ACCURACY** |  |  |
|  | Visitor deduplication | No double-counting |
|  | Metric consistency | Article-level matches site-level rollups |
|  | Test coverage | 100% pass rate on known scenarios |
|  | Data validation | Stakeholder validation: Numbers make sense |
| **USER EXPERIENCE** |  |  |
|  | Metric clarity | Stakeholders understand “Unique Reach” |
|  | Documentation | Dashboard has clear explanations and tooltips |
|  | Business value | Metrics answer key business questions |
|  | User satisfaction | Positive feedback from stakeholders |

## 8. Technical Implementation

### Solution Architecture

The technical solution involves three main steps:

1. **Define time windows** for each article based on publishing date
2. **Flag visitors** who visited during any article’s relevant time window
3. **Aggregate** using DISTINCT counts to deduplicate visitors

### SQL Implementation

-- ==============================================================================  
-- STEP 1: Define Time Windows for Each Article  
-- ==============================================================================  
-- This CTE creates the date boundaries for each article's "first X days" periods  
-- For example: Article published on Jan 1 has First 7 Days = Jan 1 to Jan 7  
  
WITH article\_timeframes AS (  
 SELECT  
 p.page\_id,  
 p.site\_id,  
 p.publishing\_date,  
 -- Calculate the end date for each time period  
 -- Using DATEADD to add days to publishing date  
 -- Note: We add 6 days (not 7) because publishing date is day 0  
 DATEADD(day, 6, p.publishing\_date) AS end\_date\_7d,  
 DATEADD(day, 13, p.publishing\_date) AS end\_date\_14d,  
 DATEADD(day, 20, p.publishing\_date) AS end\_date\_21d  
 FROM pages p  
 WHERE p.publishing\_date IS NOT NULL -- Only include published articles  
),  
  
-- ==============================================================================  
-- STEP 2: Flag Visitor Activity Within Time Windows  
-- ==============================================================================  
-- For each visitor, determine if they visited each article during its  
-- "first X days" window. This creates binary flags (1 or 0) per visitor.  
  
visitor\_in\_timeframe AS (  
 SELECT  
 a.site\_id,  
 v.visitor\_id,  
 v.division,  
 v.region,  
 a.page\_id, -- Keep page\_id for detailed analysis if needed  
  
 -- Flag: Did this visitor view this article in its first 7 days?  
 -- MAX is used in case a visitor had multiple sessions  
 -- Returns 1 if they visited at least once in the window, 0 otherwise  
 MAX(CASE  
 WHEN v.visit\_date BETWEEN a.publishing\_date AND a.end\_date\_7d  
 THEN 1  
 ELSE 0  
 END) AS visited\_in\_first\_7d,  
  
 -- Flag: Did this visitor view this article in its first 14 days?  
 MAX(CASE  
 WHEN v.visit\_date BETWEEN a.publishing\_date AND a.end\_date\_14d  
 THEN 1  
 ELSE 0  
 END) AS visited\_in\_first\_14d,  
  
 -- Flag: Did this visitor view this article in its first 21 days?  
 MAX(CASE  
 WHEN v.visit\_date BETWEEN a.publishing\_date AND a.end\_date\_21d  
 THEN 1  
 ELSE 0  
 END) AS visited\_in\_first\_21d  
  
 FROM article\_timeframes a  
 INNER JOIN visits v  
 ON a.page\_id = v.page\_id  
 GROUP BY  
 a.site\_id,  
 v.visitor\_id,  
 v.division,  
 v.region,  
 a.page\_id  
),  
  
-- ==============================================================================  
-- STEP 3: Deduplicate Visitors Across Articles  
-- ==============================================================================  
-- A visitor might have visited multiple articles in their respective windows.  
-- We need to collapse this to one row per visitor, flagging if they visited  
-- ANY article in its first X days.  
  
visitor\_deduplicated AS (  
 SELECT  
 site\_id,  
 visitor\_id,  
 division,  
 region,  
  
 -- If a visitor viewed ANY article in its first 7 days, flag them  
 -- MAX collapses multiple article visits into a single flag per visitor  
 MAX(visited\_in\_first\_7d) AS reached\_in\_first\_7d,  
 MAX(visited\_in\_first\_14d) AS reached\_in\_first\_14d,  
 MAX(visited\_in\_first\_21d) AS reached\_in\_first\_21d,  
  
 -- Also calculate total interactions (for hybrid approach)  
 SUM(visited\_in\_first\_7d) AS interactions\_first\_7d,  
 SUM(visited\_in\_first\_14d) AS interactions\_first\_14d,  
 SUM(visited\_in\_first\_21d) AS interactions\_first\_21d  
  
 FROM visitor\_in\_timeframe  
 GROUP BY  
 site\_id,  
 visitor\_id,  
 division,  
 region  
)  
  
-- ==============================================================================  
-- STEP 4: Final Aggregation with Filters  
-- ==============================================================================  
-- Apply user filters and calculate both Unique Reach and Total Interactions  
  
SELECT  
 site\_id,  
  
 -- METRIC 1: Unique Reach (deduplicated visitors)  
 -- Count each visitor only once, even if they visited multiple articles  
 COUNT(DISTINCT CASE  
 WHEN reached\_in\_first\_7d = 1  
 THEN visitor\_id  
 END) AS unique\_visitors\_first\_7d,  
  
 COUNT(DISTINCT CASE  
 WHEN reached\_in\_first\_14d = 1  
 THEN visitor\_id  
 END) AS unique\_visitors\_first\_14d,  
  
 COUNT(DISTINCT CASE  
 WHEN reached\_in\_first\_21d = 1  
 THEN visitor\_id  
 END) AS unique\_visitors\_first\_21d,  
  
 -- METRIC 2: Total Interactions (duplicates allowed)  
 -- Sum up all article-level interactions  
 SUM(interactions\_first\_7d) AS total\_interactions\_first\_7d,  
 SUM(interactions\_first\_14d) AS total\_interactions\_first\_14d,  
 SUM(interactions\_first\_21d) AS total\_interactions\_first\_21d,  
  
 -- METRIC 3: Overlap Rate  
 -- Percentage of interactions that are from repeat visitors  
 ROUND(  
 100.0 \* (  
 SUM(interactions\_first\_7d) -  
 COUNT(DISTINCT CASE WHEN reached\_in\_first\_7d = 1 THEN visitor\_id END)  
 ) / NULLIF(SUM(interactions\_first\_7d), 0),  
 2  
 ) AS overlap\_rate\_7d\_pct,  
  
 -- Count of distinct articles included  
 COUNT(DISTINCT visitor\_id) AS total\_visitors\_analyzed  
  
FROM visitor\_deduplicated  
  
-- Apply user-selected filters here  
WHERE  
 site\_id = 'News&Events' -- Example: Site filter  
 -- AND division = 'Marketing' -- Example: Division filter  
 -- AND region = 'EMEA' -- Example: Region filter  
  
GROUP BY site\_id;

### Code Explanation

#### Key Techniques Used

1. **DATEADD for Time Windows**
   * DATEADD(day, 6, publishing\_date) creates a 7-day window (day 0 + 6 days)
   * Ensures consistent time period calculation across all articles
2. **CASE Statements for Flagging**
   * Creates binary indicators (1/0) for visitor activity within windows
   * Enables flexible aggregation in later steps
3. **Deduplication Strategy**
   * First: Flag visits at article level
   * Then: Collapse to visitor level using MAX()
   * Finally: Count DISTINCT visitors at site level
4. **Hybrid Metrics**
   * Unique Reach: Uses COUNT(DISTINCT visitor\_id)
   * Total Interactions: Uses SUM() of flags
   * Both calculated from same dataset for consistency

### Performance Optimization Notes

For large datasets, consider these optimizations:

-- Option 1: Pre-aggregate at daily level  
-- Create a summary table with daily visitor-article-site combinations  
-- This reduces the data volume significantly  
  
CREATE TABLE daily\_article\_visits\_summary AS  
SELECT  
 visit\_date,  
 page\_id,  
 site\_id,  
 division,  
 region,  
 COUNT(DISTINCT visitor\_id) AS unique\_visitors  
FROM visits  
GROUP BY visit\_date, page\_id, site\_id, division, region;  
  
-- Option 2: Partition by publishing date ranges  
-- For very large datasets, partition the calculation by month/quarter  
-- Then union the results  
  
-- Option 3: Materialized view for common filters  
-- If certain filters (e.g., site-level) are queried frequently,  
-- pre-calculate and store results in a materialized view

### Testing Strategy

-- Test 1: Verify deduplication works correctly  
-- Expected: Same visitor visiting 2 articles should count as 1 unique visitor  
  
WITH test\_data AS (  
 SELECT 'V1' AS visitor\_id, 'A1' AS page\_id, '2023-01-02' AS visit\_date  
 UNION ALL  
 SELECT 'V1', 'A2', '2023-02-16' -- Same visitor, different article  
)  
-- Run main query and verify unique count = 1  
  
-- Test 2: Verify time window boundaries  
-- Expected: Visit on day 7 should be included, day 8 should not  
  
-- Test 3: Verify filter combinations  
-- Expected: Site + Division filter should return intersection, not union

## 9. Migration Plan

### Phase 1: Validation (2 weeks)

* Implement SQL in test environment
* Run parallel calculations with old and new logic
* Validate with sample of known articles
* Get business stakeholder approval on sample data

### Phase 2: Dashboard Update (1 week)

* Update dashboard to show both metrics
* Add tooltips explaining each metric
* Create user documentation

### Phase 3: Training (1 week)

* Train stakeholders on new metrics
* Provide examples and use cases
* Gather feedback

### Phase 4: Production Rollout (1 week)

* Deploy to production
* Monitor performance
* Address any issues

## 10. Open Questions for Business Discussion

1. **Primary Metric:** Should “Unique Reach” or “Total Interactions” be the primary/default metric displayed?
   * **Recommendation:** Unique Reach (given 130K employee base context)
2. **Naming Convention:** What should we call these metrics in the dashboard to avoid confusion?
   * Options for deduplication: “Unique Employees”, “Employee Reach”, “Distinct Employees”
   * Options for interactions: “Total Visits”, “Article Views”, “Engagement Volume”
   * **Recommendation:** Use “Unique Employees” and “Total Visits” for clarity
3. **Percentage Display:** Should we always show “% of Company” alongside absolute numbers?
   * Example: “3,120 employees (2.4% of company)”
   * **Recommendation:** Yes, for executive dashboards
4. **Division Denominators:** For division filters, should we show “% of Division” if we have headcount data?
   * Example: “1,250 Marketing employees (25% of Marketing division)”
   * Requires: Division headcount data
5. **Historical Data:** Should we recalculate historical data with the new logic, or start fresh?
   * Impact: Historical trends will change if recalculated
6. **Thresholds:** Are there minimum thresholds we should apply?
   * Example: Exclude articles with < 10 employees in first 7 days?
   * Example: Flag sites reaching < 0.5% of company as “low engagement”
7. **Benchmark Targets:** What are realistic employee engagement targets?
   * Industry standard for intranets: 5-15% monthly active users
   * Should we set targets like: “Goal: 5% of employees in first 7 days”?
8. **Export Requirements:** How should these metrics appear in exported reports (Excel, PDF)?
   * Include both absolute numbers and percentages?
   * Include employee segment breakdowns?
9. **Anonymous vs Authenticated:** Are all 130K employees authenticated when visiting?
   * If yes: Can we track individual employee journeys for better insights?
   * If no: How do we handle anonymous visitors?
10. **Employee Churn:** How do we handle employees who leave the company?
    * Should historical metrics be adjusted for changing headcount?
    * Example: “Reached 10% of company (based on 125K employees at that time)”

## 11. Success Criteria

The implementation will be considered successful when:

* Stakeholders can clearly interpret “First X Days” metrics at all aggregation levels
* No confusion about visitor deduplication
* Metrics are consistent across article and site levels
* Dashboard performance remains acceptable (< 5 second load time)
* Stakeholders can answer their key business questions using the provided metrics

## Appendix A: Glossary

| Term | Definition |
| --- | --- |
| **UV (Unique Visitor)** | A distinct employee who visited at least one page, identified by employee/visitor\_id |
| **Unique Employees** | Deduplicated count of distinct employees who engaged with content |
| **Publishing Date** | The date when an article was first made publicly available on the intranet |
| **First X Days** | The time period starting from publishing date (day 0) through day X-1 |
| **Site** | A collection of related pages/articles (e.g., “News & Events”, “HR Updates”) |
| **Division** | Organizational division of the employee (e.g., “Marketing”, “Sales”, “Engineering”) |
| **Region** | Geographic region of the employee (e.g., “EMEA”, “North America”, “APAC”) |
| **Deduplication** | Removing duplicate employee counts when aggregating across articles to count each person once |
| **Overlap Rate** | Percentage of visits from employees who viewed multiple articles during their launch periods |
| **Employee Base** | Total company population of 130,000 employees |
| **% of Company** | Unique employees reached as a percentage of the 130,000 employee base |
| **Total Interactions** | Sum of all article visits (same employee counted multiple times if they read multiple articles) |
| **Unique Reach** | Count of distinct employees (each employee counted once regardless of how many articles they read) |

## Appendix B: Comparison Table

| Aspect | Option 1: Total Interactions | Option 2: Unique Reach | Option 3: Hybrid |
| --- | --- | --- | --- |
| **Calculation Complexity** | Simple | Moderate | Moderate |
| **Interpretation Clarity** | Easy | Requires explanation | Moderate |
| **Business Value** | Measures engagement volume | Measures audience size | Both |
| **Comparison to Article Level** | Direct (sum) | Indirect | Both |
| **Risk of Misinterpretation** | High (inflates %) | Low | Low |
| **Shows % of Company** | No (misleading) | Yes (accurate) | Yes |
| **Development Effort** | 1 week | 2 weeks | 2.5 weeks |
| **Query Performance** | Fast | Fast (130K user base) | Fast |
| **Stakeholder Training Needed** | Minimal | Moderate | Moderate |
| **Recommended for 130K Employees** | ❌ | ✅✅ | ✅✅✅ |

## Appendix C: Technical Advantages of 130K Employee Base

The fact that this is an internal intranet with 130,000 employees (not an external website) provides **significant technical advantages**:

### Performance Benefits

| Aspect | Internal (130K employees) | External (unlimited users) |
| --- | --- | --- |
| **Maximum visitor\_ids to track** | 130,000 | Unlimited (millions/billions) |
| **Deduplication memory** | ~2-5 MB | Could be GBs |
| **Query performance** | Very fast (small lookup table) | Can be slow (huge lookup) |
| **Pre-aggregation feasibility** | Easy (can pre-calc all combos) | Difficult (too many combinations) |

### Specific Technical Wins

**1. Visitor Lookup Table Fits in Memory**

130,000 employees × 16 bytes per ID = 2 MB  
Even with metadata: < 10 MB total  
  
Result: Entire employee base fits in Databricks executor memory  
→ Ultra-fast joins and deduplication

**2. Can Pre-calculate All Division/Region Combinations**

130,000 employees across:  
- 20 divisions  
- 10 regions  
- 774 sites  
  
All possible combinations: Manageable to pre-calculate  
Storage required: < 100 MB  
  
Result: Sub-second dashboard queries for any filter combination

**3. Simplified Data Model**

-- Employee dimension table (small, static)  
CREATE TABLE employees (  
 employee\_id STRING,  
 division STRING,  
 region STRING,  
 -- ... other attributes  
) -- Only 130K rows!  
  
-- Can use broadcast joins for maximum performance  
SELECT /\*+ BROADCAST(employees) \*/ ...

**4. Realistic Testing**

With 130K employee base, you can:  
- Generate realistic test data (sample 10K employees)  
- Validate metrics against known ground truth  
- Run end-to-end tests in minutes, not hours

### Query Performance Estimates (Updated for 130K Employees)

| Query Type | Estimated Performance | Confidence |
| --- | --- | --- |
| Single site (any filter) | < 0.5 seconds | ✅ Very High |
| Multiple sites | < 1 second | ✅ Very High |
| Complex multi-filter | 1-3 seconds | ✅ High |
| Full company rollup | < 1 second | ✅ Very High |
| Historical trend (3 years) | < 2 seconds | ✅ High |

**Why so fast?** - Employee lookup table is tiny (130K rows vs millions/billions) - All deduplication happens in-memory - Can use broadcast joins everywhere - Pre-aggregation tables are < 50 MB total

### Storage Requirements (Updated)

| Component | External Website (est.) | Internal Intranet (130K) |
| --- | --- | --- |
| Raw visit logs | 500 GB - 5 TB | 20-50 GB |
| Visitor lookup table | 1-10 GB | **< 5 MB** ✅ |
| Pre-aggregated metrics | 100 GB - 1 TB | **< 100 MB** ✅ |
| Total storage | Multi-TB | **< 100 GB** ✅ |

**Cost impact:** 10-100x cheaper storage and compute costs

**Document Version:** 1.0 **Last Updated:** 2025-10-09 **Author:** Data Engineering Team **Review Status:** Draft - Pending Business Approval