Adobe© Marketing Cloud

Insight with SiteCatalyst Implementation Guide

Contents

General Differences between Insight and SiteCatalyst

Adobe SiteCatalyst and Adobe Insight consume and interact with data differently in a few instances when implemented using baseline configurations. To understand and reconcile these platform differences, refer to the sections below:

- **Tips for integrating Insight and SiteCatalyst**

  Basic strategies and general guidelines are listed for implementing integrated SiteCatalyst and Insight data feeds and features.

- **Data handling differences**

  For baseline implementations, SiteCatalyst and Insight occasionally differ in the way they handle data and identify time dimensions.

- **Attribution Logic**

  Attribution of time and participation data is handled differently in SiteCatalyst and Insight.

- **Insight and SiteCatalyst Interactions**

  Comparisons of Insight logic with SiteCatalyst logic for integrated interaction and functionality.

Tips for Integrating Insight and SiteCatalyst

About Baseline Configuration and Customization: For initial implementation, it is recommended that you adopt the baseline profile configuration and dimensions provided with standard SiteCatalyst and Insight setups. Customizing the respective dimensions and metrics is not recommended for implementation unless working with consulting or an understanding of both products and their implementation techniques. To compare reported data values, see the *SiteCatalyst Comparative Profile* to compare hit-level times and other dimensions of SiteCatalyst with Insight. This comparison of data values in each platform is researched and validated by Adobe engineers and defines the differences between SiteCatalyst and Insight data feeds and dimension reporting. For Insight users, the baseline workspaces for initial comparison of standard SiteCatalyst reports are accessed through the SiteCatalyst Compare workspace tab in Insight.

Differences in applying SiteCatalyst and Insight metrics. SiteCatalyst metrics applied to a report in SiteCatalyst vary depending on the type of dimension used in that report, allowing it to be consumable and understandable by the analyst. That is, the context in SiteCatalyst changes to meet report needs. For Insight, each metric is global and unspecified. As a result, when the metric is applied against any dimension, it will follow the exact same rules. Insight requires the analyst to apply the proper metrics for the data to make sense.

SiteCatalyst metrics renamed for accuracy. In SiteCatalyst 15, some metric names have been renamed to be more accurate. For example in the SiteCatalyst Prop report, you now have Instances and Page View metrics to build the report (among others). These metrics now compare with the Insight metrics. For example, Page Views changed names to Instances. For SiteCatalyst 15, you now compare it to the Instances metric in Insight.

Data Handling Differences

In the following examples, the baseline implementation of Insight and SiteCatalyst capture and report event times differently.

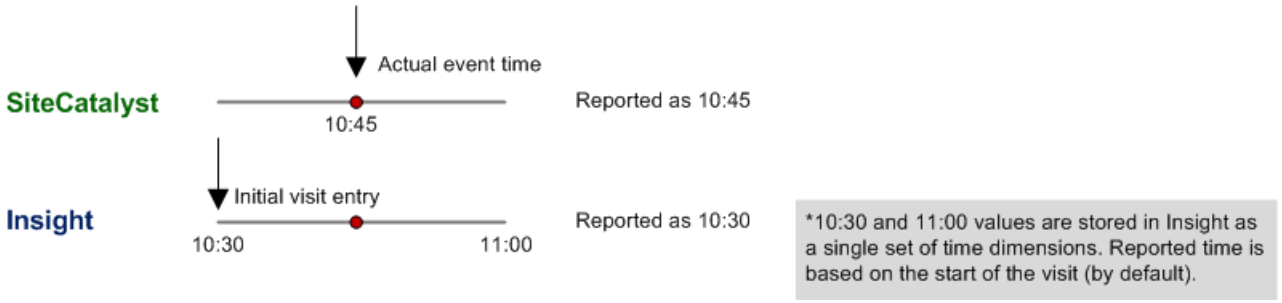
Although these discrepancies occur for baseline configurations of each product, Insight also includes functionality to duplicate or approximate all standard or extended SiteCatalyst features—including the real-time reporting of

events. Capturing a set of time dimensions for events at the start of the session is a standard Insight best practice, but you can add customized features as needed.

These differences occur only in baseline Insight configurations.

Event times captured differently

For SiteCatalyst, the time of an event is captured and reported within one minute of its occurrence. In contrast, Insight creates a single set of time dimensions based upon the start of the visit. As a result, Insight attributes all event times to the initial entry time of the visit, rather than exactly when the event occurred.



Because of this time reporting discrepancy, broader time period settings mean better alignment between SiteCatalyst and Insight event reporting. For example, a monthly setting would mean a lower percentage of error—perhaps less than a 1 percent discrepancy. But for time period settings of lower granularity, such as an hour, the reporting data is larger so the discrepancies will be much greater—perhaps as much as 50 percent.

If an evar is configured with linear allocation in SiteCatalyst, then Insight will not be able to support it with a similar feature (this will be a rare occurrence). However, although linear allocation support is not part of the Insight baseline configuration, these features can be added or approximated through custom alterations in Insight. Insight implementations can reproduce most of the allocation methods passed from SiteCatalyst using custom features.

Visitor profile data limited by record size in Insight

Discrepancies exist in handling visitor data between Site Catalyst and Insight.

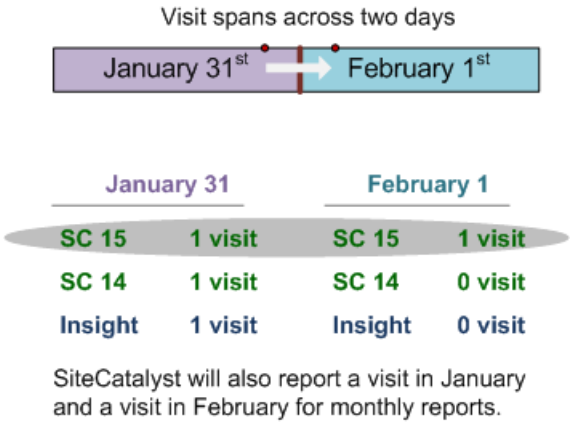
When the data exceeds the limit of data stored for a visitor record, a second visitor ID and record is generated. This duplicates visitor records and elevates the unique visitor count and associated metrics. Data sets with high numbers of repeat visitors and large row sizes will have a higher number of splits due to these limitations. Consequently, although order numbers are not be affected, some secondary attributions will be incorrectly evaluated.

For example, although the number of orders will not differ, the attributed orders to each campaign may be affected. This is due to a new Visitor record being generated without an actual new visitor, which is then evaluated as part of the campaign ratios.

Note: *Although the visitor profile data setting is configurable, it is recommended to leave it at the default value unless working with engineering or consulting experts. If record size is set to a high value, then data throughput will become an obstacle.*

Time reporting

Visits that span across distinct time frames—such as starting on one day or month and ending on the next—will generate two Visit events when reporting in SiteCatalyst 15. In SiteCatalyst 14 or versions of Insight, these visits are counted only for the beginning time of each visit.



In this example, SiteCatalyst 15 will also count a visit for a January report, as will SiteCatalyst 14 and Insight.

Note: *For Adobe Discover, time reporting is the same as SiteCatalyst 15.*

Unique values exceeded

SiteCatalyst and Insight have different ways of handling element uniqueness (cardinality) data when it exceeds data size limits. A Unique Exceeded message will appear when either limit is met.

- SiteCatalyst allows 500 K unique values per month before returning a "uniques exceeded" message. This value is configurable and based on different dimension types.
- Insight defaults to 32 K unique values per dimension based on all data loaded using a Simple dimension. If using a Denormal dimension, you can report all Insight data by displaying fewer events (shows 1024 elements per report). Once that record size is exceeded, the dimension saves the additional Unique Exceeded values into a secondary "Small Elements" table.

  In Insight, the first 32 K of elements are captured and synched with the 32 K of SiteCatalyst elements. See the Insight guides for additional information.

Note: *The Unique Exceeded elements identified in SiteCatalyst will not compare directly with Insight elements. Some elements will display in SiteCatalyst that are not in Insight. And conversely, some values in Insight will not display in SiteCatalyst.*

Case comparison handling in SiteCatalyst and Insight
Rules handling the varying case of characters with the same terms is handled differently in SiteCatalyst and Insight:

- Insight ignores case as standard logic. It also allows values to be aggregated by case differences to reduce duplicated cardinality readings.
- SiteCatalyst has mixed rules regarding case depending on the dimension type.

Attribution Logic Differences

SiteCatalyst and Insight handle assigned attribution values differently in some instances.

The following are logic differences for integrated Insight and SiteCatalyst attribution.

Participation

Linear allocation

Participation

Insight does not support the SiteCatalyst Participation dimension directly, but includes a Visit-level segmentation dimension to approximate this SiteCatalyst functionality. By design, Insight captures any dimension values that occur within that session. As a result, when you build a Visit-level segment in Insight, these values are included in the Visit-level segment and reported as part of that event. This comprehensive data set allows you to not only identify the purchases, but allows you to identify pages hit before and after the visit.

In contrast, SiteCatalyst participation metrics will not include other page hits or events actuated during the session, and will attribute revenue only to the transaction pages.

For example: When a visitor views a Product Information page, Insight allows you to identify everything that occurred during the visit, including views of additional pages. Revenue is distributed across all pages that participated in the purchasing session.

Revenue is attributed to each page based on participation logic.

SiteCatalyst	
A	$100
B	$100

Insight	
A	$100
B	$100
C	$100

SiteCatalyst will only identify participation for those purchase pages directly interacting with the customer's purchase. Consequently, SiteCatalyst will only attribute revenue amounts for the pages where the product was purchased. For Insight, the full revenue is distributed to all pages as part of the visit.

Linear allocation

If an Evar is configured with linear allocation in SiteCatalyst, then Insight will not be able to support it with an exact corollary feature (this will be a rare occurrence). However, although linear allocation support is not part of the Insight

baseline configuration, these features can be added or approximated through custom alterations in Insight. Insight implementations can reproduce most of the allocation methods passed from SiteCatalyst using custom features.

Important: Custom dimensions with a selected configuration for the client's needs can be set up using custom features. Linear allocation is not deployed by default on baseline dimensions, but can be customized by the customer or through consulting. In part because certain fields are best built at a particular level in the Insight tool. However, while the baseline configuration attempts to use the most common and generally accurate method of configuration for dimensions, it still may not match the logic applied in SiteCatalyst. The customer needs to choose the most important aspects of the dimension for their business needs, and then apply the appropriate configuration to make those aspects of the dimension accurate.

Insight and SiteCatalyst Interactions

The following issues compare Insight with SiteCatalyst to identify how data feeds and logic interact with each other.

The following are basic interactions between Insight and SiteCatalyst:

Evars

Original Referring Domain

Referring Domain

Evars

Insight separates the instance instead of the visit-attributed refer. Through initial validation, single dimensions contain the data for both Instance and Persisted (attributed to revenue) data. However, for future versions of Insight 5.5, two dimensions will be used when comparing Insight to the SiteCatalyst reports.

For example, Evars will now have the following dimensions:

• Evar# — The instances when the value is set or encountered during the visit.
• Conversion Evar# — The persisted Evar value that exists on the rows with Revenue (commerce data).

Original Referring Domain

Insight extracts the full, original reference value of the domain name (from the First Hit Refer column). SiteCatalyst reports the top-level domain. For example in mail.adobe.com, the original referring domain is adobe.com.

In contrast, Insight derives the top-level domain through logic from the full domain name. If the data feed contains the following value:

`http://www.google.com/search?client=firefox-a&rls=org.mozilla%3Aen-US%3Aofficial&channel=s&hl=en&source=hp&biw=1534&bih=956&q=web+analytics&btnG=Google+Search`

Then Insight will derive the top-level domain based on the value provided in `first_hit_referrer` from the data feed to return `google.com` as the original referring domain.

Referring Domain

In Insight, configurable dimensions provide the functionality to capture and report data similar to the reports in SiteCatalyst . For example, Insight uses these Dimensions to return the name of the referring web page previous to a purchase. Insight reports can be set to reproduce SiteCatalyst reports:

• Referring Domain Instance—Instances when the referring domain is set or encountered during the visit.

• Referring Domain—Referring domain value for the first row of the visit.

Being at the Visit level, these dimensions will align and interact with the Commerce dimensions to identify participation in the purchase.

Common Insight with SC Implementation Features

This section identifies integrated Insight and SiteCatalyst features for the user implementation.

The features and implementation tasks listed below also reference Insight documentation for complete feature information. For documentation resources identified in this section, go to *http://microsite.omniture.com/t2/help/en_US/home/index.html#Insight*

SiteCatalyst Clickmap matched with Insight Dimensions and Metrics

Identify the dimensions and metrics used by Insight to match raw SiteCatalyst clickmap data.

You will use the **Click Context** and **Click Action** along with the **Instance** metric to compare against SiteCatalyst. The Click dimension is located under `Site Content\Link\Click Map`.

Change metrics on Clickmap overlay

You can change metrics on the **Clickmap** overlay to render information about your data on the image of a web page. This type of visualization illustrates how individual links on the web page relate to specific dimensions and elements within your data set. Page overlays, like all visualizations, are generated from queries of the data set.

The utility of a page overlay depends entirely upon the accuracy of the mappings of the links to the underlying data. If the links are mapped incorrectly or inappropriately, it can be misleading. For high-stakes analyses or cases when you need precise quantitative results, consider making more explicit queries.

For complete information on changing metrics, see the Page Overlays section (5.9) of the *Insight User's Guide for Clients*.

Segments by strict path order

How to create segment based on a strict A > B > C path order.

A path browser enables you to analyze the sequence in which a particular dimension’s elements were accessed. You create path browsers by dragging and dropping an element of a dimension onto a blank path browser visualization. The element that you drag and drop onto the path browser becomes the “root” of the path browser. The path browser displays “paths” that pass through the root, enabling you to see the sequence of elements that were accessed before and after the root.

For the many implementations, you will want to make the appropriate path selection within a workspace using the path browser. Then add a segment to a Visit level segment visualization.

For additional information, see Path Browsers section (5.10)

Customized page, site section, and video count for a Visit

Customizing dimensions and metrics for counting pages, sections, and videos viewed in a Visit session.

See *Creating and Editing Derived Metrics* in the *Insight 5.5 User Guide* to create metrics, and refer to the *Syntax for Metric Expressions* in Appendix B for syntax reference details.

Previous and Next dimension shifts at multiple levels

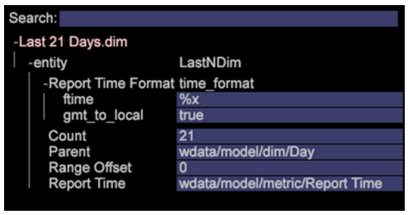
To create negative and positive shift dimension for pages, sections, and videos, see the *Creating Shift Dimensions for Multiple Levels* section of this document. This section details the creation and modification of client-side Shift dimensions.

Identifying Last 30 days dimension

Create a dimension in Insight and document how to create a "day" dimension based on a customized number of days.

Modify an existing Reporting dimension found in a Profile Manager under Dimensions\Time\Reporting.

1. Make file local and rename it. For example: "Last 21 Days.dim."
2. Open the file in the Workstation.
3. Update Count to cover the desired time period. Based on example change Count: 21 to 30 will produce a "Last 30 Days" dimension.



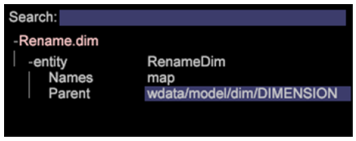
Loading sample data set

Log processing data set includes files that contain additional instructions for the log processing phase of dataset construction. These files exist within the Dataset\Log Processing directory for any inherited profile, and they typically define application-specific parameters, such as web-specific configuration parameters.

In the *Insight Dataset Guide*, refer to the *Editing the Log Processing Configuration File* in section 2.2 for information pertaining to the "Hash Threshold" parameter.

Setting up client-side classifications

Create a Client-Side Rename dimension based on the extended dimension that has elements that need to be mapped. An example can be found in the Profile Manager under "Dimensions\Rename.dim.example".



Import/Export Visitor ID segments

Visitor ID segments are set up on a case-by-case basis during each implementation, and pertain to Segment Exports and the data set configuration. See the *Configuring Segments for Export* section (8.12) in the *Insight 5.5 User Guide*.

Dumping current data set and load historical data

Refer to the *Insight Dataset Guide* in the Log Files section (2.3.2) for information related to modifying log sources. Update the appropriate Log Path, save the configuration file, and save it to the profile. This will initiate a reprocess that drops the current data set and builds a new one based on the specified source data.

Setting up report delivery

For all implementations, it is important to work with the customer to define how to modify the definition and delivery of reports. See the *Insight Report Guide* section, *Working with Reports*.

Pausing and restarting nightly transforms

The Log Processing Mode.cfg file enables you to pause processing of data into a data set, specify offline sources, or specify the frequency at which Insight Server saves its state files.

See the *Additional Configuration Files* section (1.1.2) in the *Insight Dataset Guide* for additional information.

Differences between Path Browser and Next Page reports

Path Browser and the Next Page dimension work on the same premise. They shift from the current countable record to the next countable and represent information about that records. The typical example is Page Name. See*Creating Shift Dimensions for Multiple Levels* for additional information.

Differences between Attributed and Instance-level Referrer dimensions

Attributed Referrer dimension are configured for comparison with SiteCatalyst commerce information (revenue, units, orders, etc.)

Instance Referrer dimensions are configured for comparison with SiteCatalyst traffic and interaction information (instances and visits)

The At Least One selection in Mask

The Mask > At Least One setting is only available within a *Denormal* dimension. Applying an *At Least One* selection enables the Denormal dimension to re-declare the top 1024 elements based on the current workspace selections.

Are custom benchmarks created?

No, the benchmarks are based on the data set expected results.

Path by traffic source

What dimensions are used for the next/prev page?

A *Shift* dimension can be configured leveraging the desired *Traffic Source* dimension. But what is the nature of this request? This type of direct pathing isn't typically desired when talking about Traffic Sources. In most cases, the analysis around what other traffic sources visitors interact with over their lifetime or during a visit without the concept of direction, before or after, is performed.

This type of analysis can be performed with segments, filters or process maps.

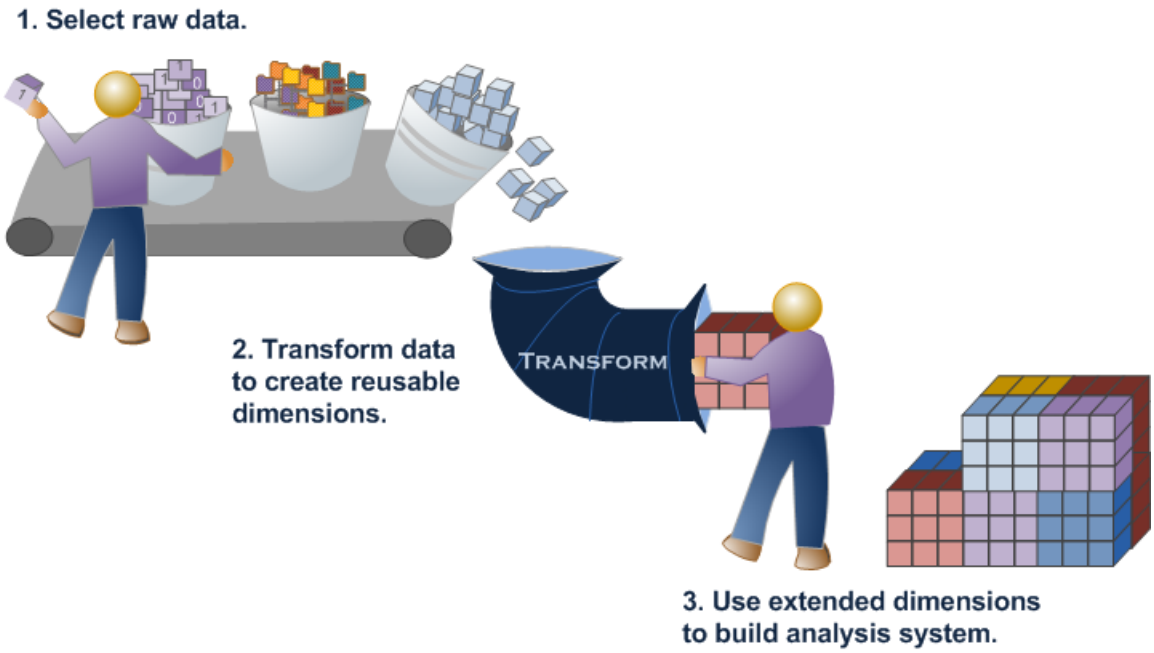
Reprocessing SiteCatalyst Data in Insight

Insight will reprocess SiteCatalyst data for reports and visuals and add custom events for many implementation requirements.

During dataset construction, InsightServer reads source data from log sources, applies transformations to specific fields of data, and defines extended dimensions to be created from the transformed fields. The construction process occurs in two phases: log processing (see “Log Processing”on page8) and transformation (see “Transformation”on page9). After the dataset is constructed, you can use the dataset’s extended dimensions to create derived metrics and dimensions for your specific analysis purposes.

See

Dataset Construction: Dataset construction is like a manufacturing process. You select the data (the raw materials) to be used to build the data set, and then you define the data transformations (the process steps) that manipulate the information available in the data to create extended dimensions (the manufactured products).



Specifying Log Sources. Log sources are files that contain the data to be used to build a data set. The data available in the log sources is called event data because each data record represents a transaction record or a single instance of an event. In addition, each record, or log entry, contains a value referred to as a tracking ID.

A transformation is a set of instructions that you can define to extract or manipulate information in the event data. Each transformation that you define is applied to each event data record (log entry) to update existing log fields or produce new fields. The results of transformations are used along with log entry conditions to evaluate which log entries will be filtered out of the data set during log processing.

Customer Reprocess and addition of custom event 34 & 35

Files modified:

- `Dataset\Log Processing\Decoding Instructions.cfg`
- `Dataset\Log Processing\SC Fields.cfg`
- `Dataset\Transformation\Custom Conversion\Custom Events.cfg`

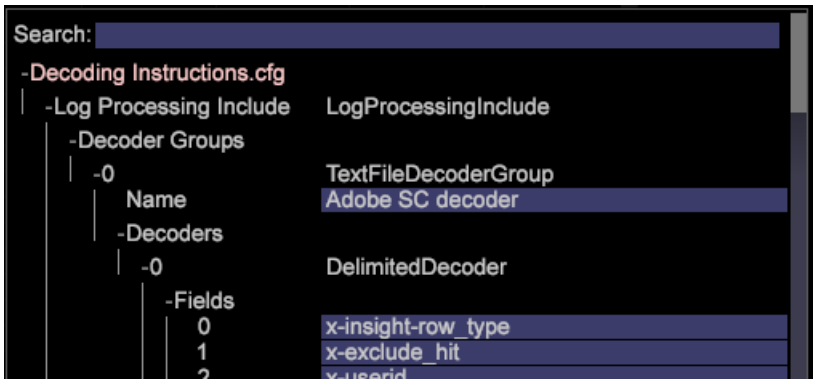
Declaring Fields in Decoder

1. Navigate to the Dataset\Log Processing\Decoding Instructions.cfg from within a Profile Manager.
    1. Decoding Instructions.cfg located under the "Adobe SC" column should be used as a reference. This file contains **all** the field declarations related to the SC data.
    2. Decoding Instructions.cfg located under the Customer column is the one that needs to be modified. This file decodes only the Customer relevant fields from the SiteCatalyst data.
2. Add the new fields to the decoder.
    a. Right-click on Decoding Instructions.cfg located under "Adobe SC" and make it local.

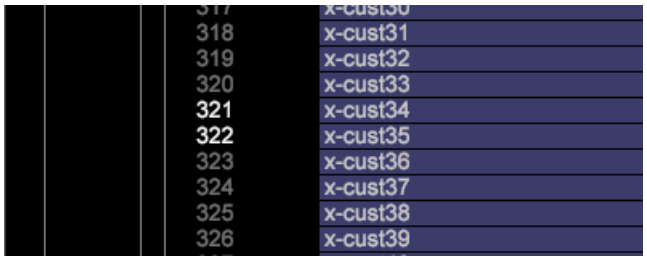
b. Right-click on the check mark, corresponding to Decoding Instructions.cfg, under the User column and open in Workstation.
c. Right-click the Decoding Instructions.cfg located under "Customer" and make it local.
d. Right-click on the check mark, corresponding to the Decoding Instructions.cfg under the User column and open in Workstation

   At this point you will have two Decoding Instructions.cfg windows open side by side.

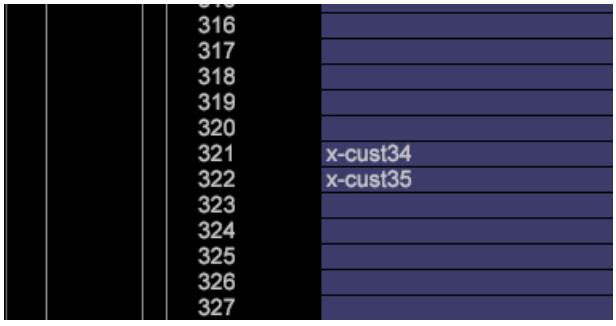
e. Expand both windows: `Decoder Groups > TextFileDecoderGroup (Adobe SC decoder) > Decoders > Delimited Decoder > Fields`



The new fields will be added under this Fields section. Use the "Adobe SC" version to locate the correct position in the decoder to add these new fields, x-cust34 and x-cust35. (Position 321 and 322 respectively)



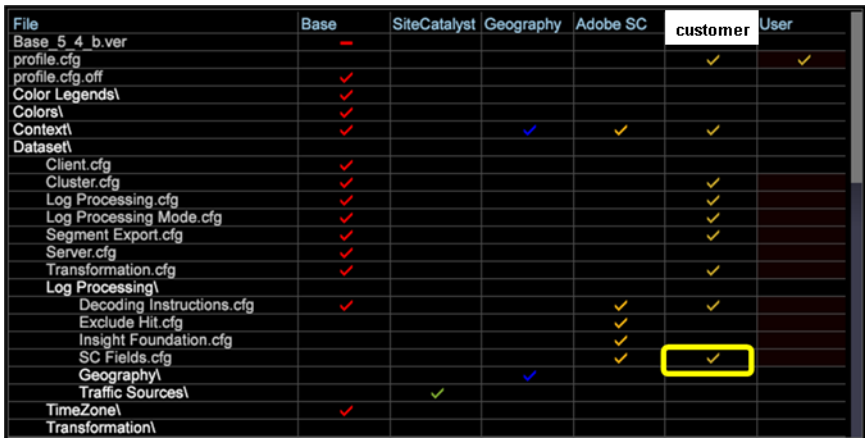
Specify these fields in the customer version of the configuration file.



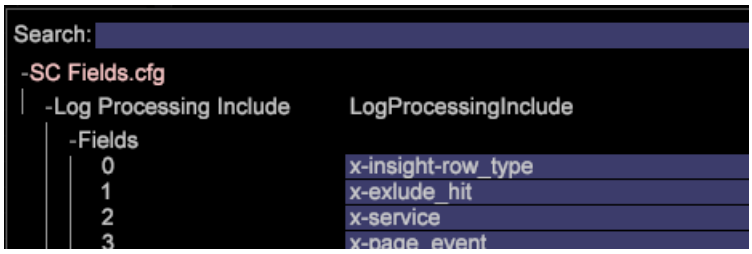
3. Save the customer version of the Decoding Instructions.cfg by scrolling back to the top and right-clicking where it says "Decoding Instructions.cfg (modified)" and selecting the first option of "Save as Dataset\Log Processing\Decoding Instructions.cfg"
4. Close both Decoding Instruction.cfg files.

Passing fields from Log Processing to Transformation

1. Navigate to the Dataset\Log Processing\SC Fields.cfg from within a Profile Manager.

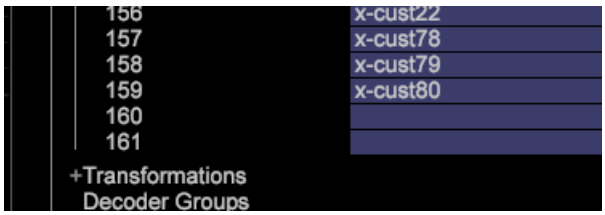


2. Right-click on the file from under the "customer name" column, select Open > in Workstation.
3. Expand fields.

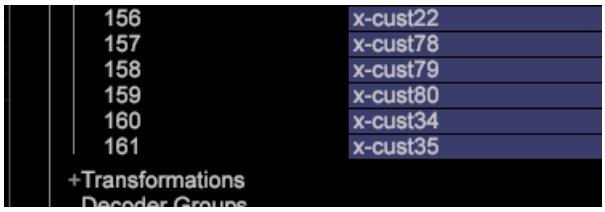


4. Scroll to the bottom of this list and perform the following action twice to make two new field declarations.

   Right-click on the number position, then select `Add new > Field`.



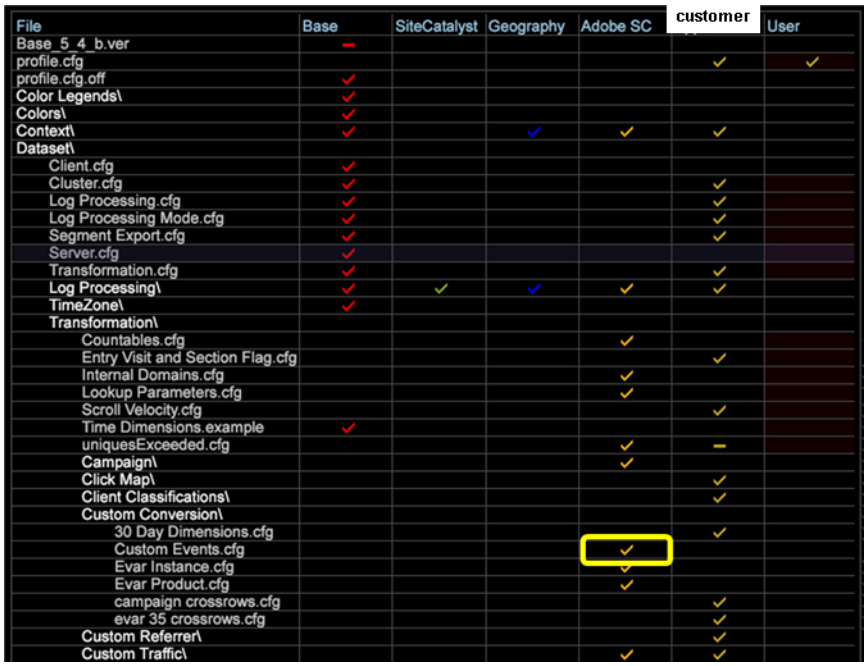
5. Type in the two new fields, x-cust34 and x-cust35.



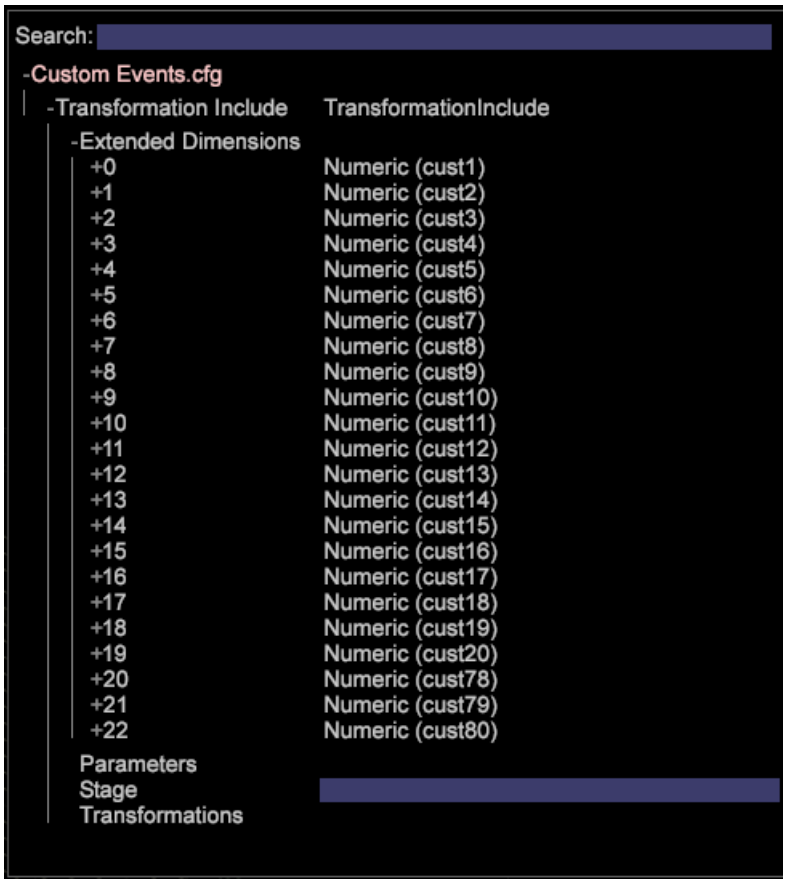
6. Save the configuration SC Fields.cfg.

Build the Dimension

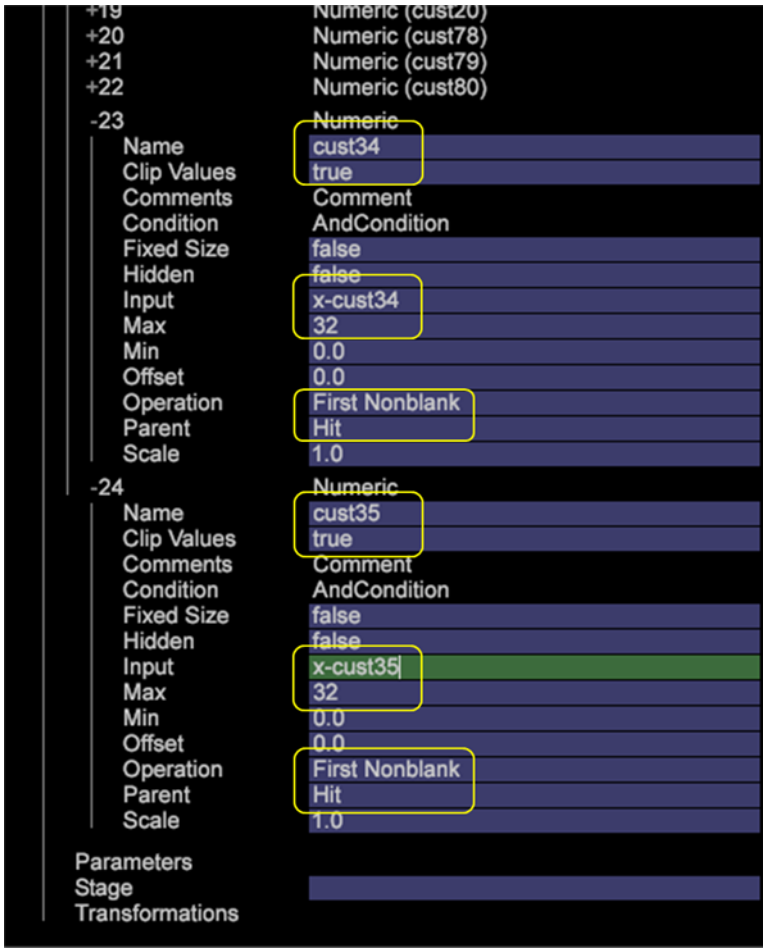
1. Navigate to the Dataset\Transformation\Custom Conversion\Custom Events.cfg from within a Profile Manager.



2. Right-click on the file and select Open > in Workstation.
3. Expand the Extended Dimension section.



4. Add the new Numeric Dimensions at the end of the Extended Dimensions list and configure them as follows (reference the highlighted yellow sections):



5. Save the configuration Custom Events.cfg.

Save changes to server

1. Save the changes to the Customer profile on the server by right-clicking the each checkmark in the User column and selecting Save to > Customer.

   Dataset\Log Processing\Decoding Instructions.cfg

   Dataset\Log Processing\SC Fields.cfg

   Dataset\Transformation\Custom Conversion\Custom Events.cfg

This will initiate the Reprocess.

Creating Shift Dimensions for Multiple Levels

Setting up Shift Dimensions for pages, sections, and videos in Insight lets your analysts capture and quantify customer interaction patterns and react with better site design and business decisions.

Implementation Issue: "When my customers interact with this new content page, what do they do next? Is it a desired result or something unforeseen?"

It's not uncommon for visitors to interact with a site or page in a different manner than was originally intended. Within Insight, it's possible to quantify where and why customers navigate in and around your site or web page by capturing

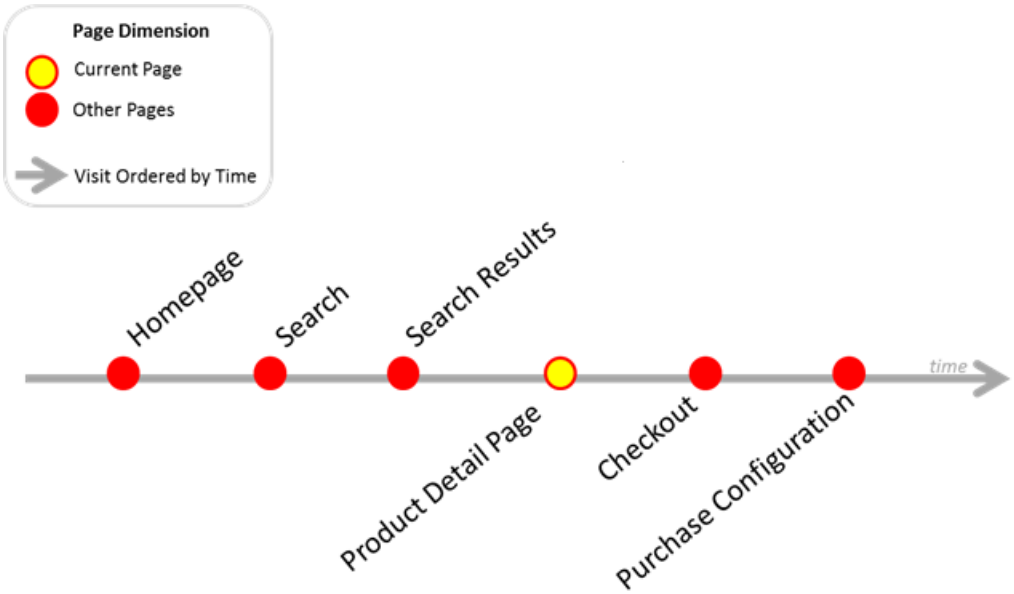
and analyzing previous and next page data. To better view visitor behavior and establish "cause and effect" relationships, Shift Dimensions were established within Insight during implementation to be used systemically. Shift dimensions act as a visitor interaction timeline to dynamically define these common navigation relationships.

About the Shift Dimension

Within the Insight dimensional model, the Shift dimension allows an analyst to look at the Nth element before or after the occurrence of that specific element within a dimension with respect to time.

The Shift dimension allows an analyst to leverage the inherent relationships available in the Insight dimensional model to dynamically investigate "cause and effect." By defining these data views from already existing dimensions, the analyst can create client-side dimensions without any impact to system sizing or processing performance.

Consider the following Visit with these Pages in time order from left to right:



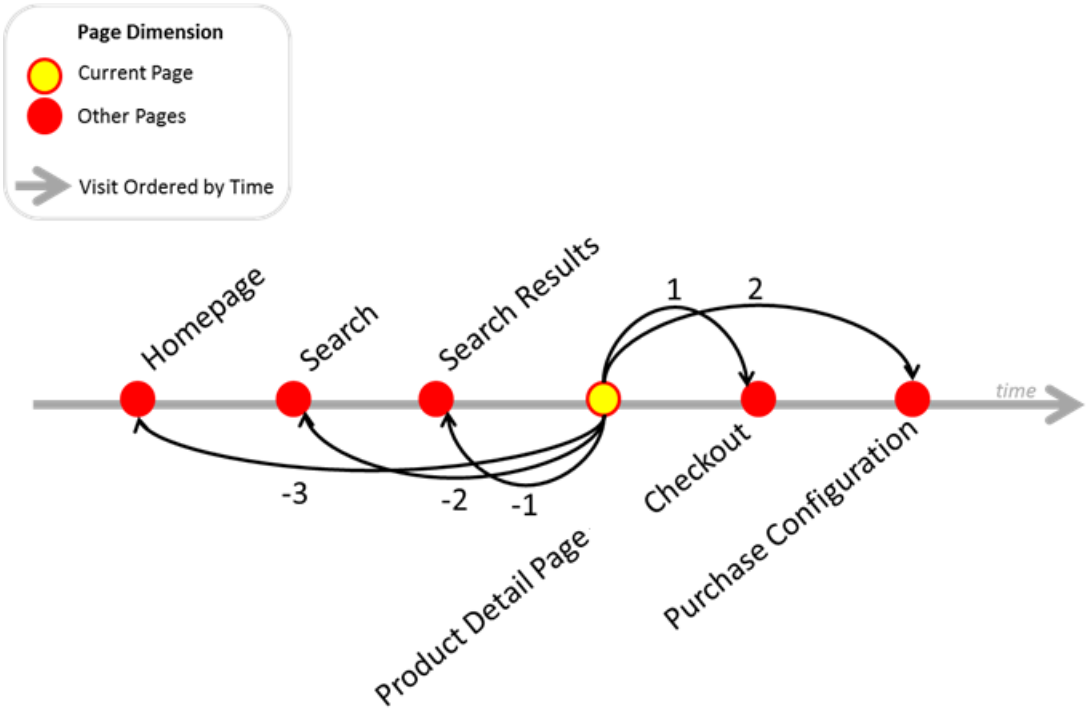
If "Product Detail Page " is the current page, what is the next page?

Checkout

What is the previous page?

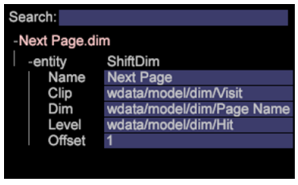
Search Results

The following diagram details how a Shift dimension establishes this relationship. The direction and distance from the current position is described by an integer value, "1" being directly next and "-1" representing directly previous.



Implementation of Shift Dimension

During implementation, the Shift dimension was set up in the Next Page dimension:



Name: Next Page

Clip: Countable level in which the relationship should be confined: Session or Visit.

Level: Countable level to be traversed and analyzed: Page View or Hit

Dim: Extended Dimension configured at the same Countable level as the defined Level. This dimension will represent the element/attribute associated with the shift.

Offset: Direction and distance to Shift from current countable Level position.

Table View for Implementation

To utilize the Next Page (Shift dimension) defined above:

Alongside a Page Name table, first apply the Page Views metric to both tables. Then select the desired starting page element from the Page Name table. The Next Page table will resolve to the appropriate page name represented on the next Hit level interaction.

Insight Dimensions and Metrics Reference

The following sections identify Product, Hit, Clickthrough, Visit, and Visitor dimensions for Insight, as well as basic metrics.

Product Dimensions

Product-based dimensions capture data about viewed and purchased site products. Subordinate product dimensions listed below are built from the parent `Product` dimension. The parent of the `Product` dimension is the `Hit` dimension.

Dimension Name	Dimension Type	Description
Product	countable	Counts the row when `x-insight-row` type is equal to 2. The parent of `Product` is `Hit`.
Products	simple	Identifies products in the data set as purchased or not purchased. This dimension uses the `x-product` field.
Viewed Products	simple	Identifies when `x-product_viewed` is empty. This dimension represents products that were viewed but not associated to a purchase or a cart. This dimension uses the `x-product` field.
Purchased Products	simple	Identifies when `x-purchaseid` is not empty. The dimension represents the products that were purchased and uses the `x-product` field.
Product Number	simple	Identifies the assigned number of the product within the purchase. This dimension uses the `x-product_num` field.
Product Count	numeric	Counts the products purchased (using inferred logic that `x-product_num` only exists on purchased item rows) for use with corresponding metrics. This is a COUNT operation using the `x-product_num` field.
Unit	numeric	Generates a sum when `x-purchaseid` is not empty. This dimension is the total number of units purchased and is used with corresponding metrics. This SUM operation using the `x-units` field.
Revenue	numeric	Generates a sum when `x-purchaseid` is not empty. This dimension is the total revenue from purchases and is used with corresponding metrics. This is a SUM operation using the `x-revenue` field.

Hit Dimensions

Hit-based dimensions capture data about internal hits to a site. Subordinate dimensions listed below are built from the parent `Hit` dimension. The parent of the `Hit` dimension is the `Visit` dimension.

Dimension Name	Dimension Type	Description
Hit	countable	Keys on the `x-hitid` field and provides a count of all rows in the system. A Hit is a row, or collection of rows, that share the same Hit ID. This allows you to associate purchases with pages or other fields where they occurred. The parent of `Hit` is `Visit`.
Purchase ID	denormal	Allows for searching by the assigned ID number and isolates any purchase in the data set using the `x-purchaseid` field.

Dimension Name	Dimension Type	Description
X-referrer-clip	denormal	Refers to each hit in the data set that does not fall into the internal domains (as configured by client and/or consulting services) using the `cs(referrer)` field. The first 244 characters of `cs(referrer)` aligns with SiteCatalyst
Clickthrough	numeric	Counts each hit that has a campaign value. It is conditioned to count when the campaign hit is not marked with a DUP value in the `x-dupe-campaign` field. It can be used to generate metrics to show the number of entries to the site that came from campaign related click-throughs. This dimension uses the `x-campaign` field.
Category	simple	Provides the first non-blank value for a given hit using the `x-category` field.
URI	simple	Identifies the HTTP request prior to any additional query name of value pairs (everything before the '?'). It is the first non-blank value for a given hit and provides the address of the page requested using the `cs-uri-stem` field.
Page Name	simple	Furnishes a plain text value for a given page using the `x-pagename` field.
Page Event Type	simple	Identifies the type of page event. It is the first non-blank value for a given hit using the `x-page_event` field.
Page Event Link Name	simple	Identifies the linking name for a page event. It is the first non-blank value for a given hit using the `x-page_event_var2` field.
Page Event Link URL	simple	Identifies the linking URL for a page event. It is the first non-blank value for a given hit using the `x-page_event_var1` field.
Page Type	simple	Identifies the page type. It is the first non-blank value for a given hit using the `x-page_type` field.
Hierarchy One	simple	Hierarchical patterns for custom use during implementation. It is the first non-blank value for a given hit using the `x-hier1` field.
Hierarchy Two	simple	It is the first non-blank value for a given hit using the `x-hier2` field.
Hierarchy Three	simple	It is the first non-blank value for a given hit using the `x-hier3` field.
Hierarchy Four	simple	It is the first non-blank value for a given hit using the `x-hier4` field.
Hierarchy Five	simple	It is the first non-blank value for a given hit using the `x-hier5` field.
Service	simple	Assists in classifying specific types of hits. It is the first non-blank value for a given hit using the `x-service` field.
Visit Page Number	simple	Identifies the ordered place of the page during the visit. It is the first non-blank value for a given hit using the `x-visit_page_num` field.
TNT combined	simple	It is the first non-blank value for a given hit using the `x-tnt` field.
Visitor Homepage	simple	Identifies if the page is a visitor homepage. It is the first non-blank value for a given hit using the `x-homepage` field.
Pages Not Found	simple	Reports when the `x-page_type` field matches an `errorPage` message and lists the pages not loading properly. It is the first non-blank value for a given hit using the `cs-uri` field.
Robot Type	simple	It is the first non-blank value for a given hit using the `x-bot_type` field.

Dimension Name	Dimension Type	Description
Robot ID	simple	It is the first non-blank value for a given hit using the `x-bot_id` field.
Reload Flag	simple	Marks rows that are reloaded using "look back" logic. If the previous `cs-uri` value is the same as the current value, then this dimension identifies it as a page reload. It is the first non-blank value for a given hit using the `x-reload-flag` field.
Carts	simple	Identifies carts per hit. It is the first non-blank value for a given hit using the `x-cart_open` field.
Cart Views	simple	It is the first non-blank value for a given hit using the `x-cart_view` field.
Cart Additions	simple	It is the first non-blank value for a given hit using the `x-cart_add` field.
Cart Removals	simple	It is the first non-blank value for a given hit using the `x-cart_remove` field.
Checkouts	simple	It is the first non-blank value for a given hit using the `x-checkout` field.
Channel	simple	Populates the dimension when the `x-service` field matches `ss`. It is the first non-blank value for a given hit using the `x-channel` field.
Exit Link	simple	Populates the dimension when `x-page_event` matches `102`, representing the link to the site to which the visitor exited. It is the first non-blank value for a given hit using the `x-page_event_var1` field.
Download	simple	Populates the dimension when `x-page_event` matches `101`, representing links to various downloads. It is the first non-blank value for a given hit. This represents links to various downloads using the `x-page_event_var1` field.
Custom Link	simple	Populates the dimension when the `x-page_event` field matches `100`. A `x-page_event_vartwo` field is created using the `x-page_event_var2` field when it is present. When empty, the `x-page_event_var1` value is copied into the field. It is the first non-blank value for a given hit using the `x-page_event_vartwo` field.
Exact Page Duration	numeric	Measures the time between hits for use in calculating visit and hit lengths. It is the first row for a given hit using the `x-exact-page-duration-ms` field.
Hit Hour, Day, Week, Month, Hour of Day, and Day of Week	time	Reports the Hit Hour, Day, Week, Month, Hour of Day, and Day of Week from the time of the 1970 epoch. These fields are built at the `Hit` level.
Order	numeric	Counts when the `x-purchaseid` field is not empty, and the `x-product_num` field is equal to 1. Because all orders have at least one product, you can infer that whenever there is a product number of 1 then the order count can be incremented by 1. This dimension provides a count of orders for use with corresponding metrics. This is a COUNT operation using the `x-product_num` field.
Props	simple	List of corresponding prop values associated with the hit.
Evars	simple	List of corresponding evar values associated with the hit where `x-insight-row_type` is equal to 1.

Clickthrough Dimensions

Clickthrough-based dimensions capture data about internal clicks to arrive at a site. The parent of the `Clickthrough` dimension is the `Visit` dimension.

Dimension Name	Dimension Type	Description
Clickthrough Dimensions	countable	Counts the row when `x-campaign` and `x-ref_domain` are not empty and where `x-insight-row_type` is equal to 1.
Keyword Clickthrough	simple	Lists keywords used to arrive at the site. It is the first non-blank value for a given click-through using the `x-keywords` field.
Referring Domain Clickthrough	simple	Lists referring domains employed to arrive at the site. It is the first non-blank value for a given click through using `x-ref_domain` field.
Referrer Type Clickthrough	simple	Lists referrer types used to arrive at the site where `x-ref_domain` is not empty. It is the first non-blank value for a given click-through using the `x-ref_type_lookup` field.
Search Engine Clickthrough	simple	Lists search engines used to find the site. It is the first non-blank value for a given click through using `x-search_engine_name` field.
Search Page Ranking Clickthrough	numeric	Lists search page ranking association for the keyword used to arrive at the site where `x-visit_search_engine_name` is not empty. It is the first non-blank value for a given click-through using `x-search_page_num` field.
Campaign Clickthrough	simple	Lists campaigns used to arrive at the site. It is the first non-blank value for a given click-through using x-campaign field.
Clickthrough Landing Page Name	simple	Lists the name of the landing page first seen when the visitor arrived at the site through a click- through. It is the first non-blank value for a given click-through using the `x-pagename` field.
Paid Search Flag Clickthrough	simple	Flags a click-through when a paid search is used to arrive at the site. It is the first non-blank value for a given click-through using `x-paid_search`.
Page Event Type Clickthrough	simple	Lists the page event type for the page when a visitor arrived at the site through a click-through. It is the first non-blank value for a given click-through using `x-page_event` field.

Visit Dimensions

Visit-based dimensions capture data about the natures of visits to a site. Subordinate dimensions listed below are built from the parent `Visit` dimension. The parent of the `Visit` dimension is the `Visitor` dimension.

Dimension Name	Dimension Type	Description
Visit	countable	Counts all visits in a system. The visit dimension keys on the `x-visit_num` field and provides a collection of rows that share the same visit number. The parent of `Visit` is `Visitor`.
Hour, Day, Week, Month, Hour of Day, Day of Week	time	Reports the Hit Hour, Day, Week, Month, Hour of Day, and Day of Week of the visit from the time of the 1970 epoch. These fields are built at the `Visit` level.
Entry Channel	simple	Populates the dimension when the `x-service` field matches the value `ss`. It represents the channel from which the visitor entered the site. It is the first non-blank value for a given visit using the `x-channel` field.
Campaign	simple	Populates identifiers of various campaigns as configured by the client. It is the first non-blank value for a given hit. This dimension uses the `x-campaign` field.
Exit Channel	simple	Represents the channel from which the visitor entered the site. It is conditioned to populate the dimension when `x-service` matches the value `ss`. It is the last non-blank value for a given visit using the `x-channel` field.
Entry Page	simple	Represents the first page accessed for each visit. It is the first non-blank value for a given visit using the `x-pagename` field.
Exit Page	simple	Represents the last page accessed for each visit. It is the last non-blank value for a given visit using the `x-pagename` field.
Referring Domain	simple	Identifies the domain of the referrer only for the given visit. It is the first row value for a given visit using the `x-ref_domain` field.
External Search Terms	many to many	Lists search phrases broken down into individual terms using the `x-search-terms` field.
Search Engine	simple	Represents the list of search engines that brought visitors to the site for a specific visit. It is the first non-blank value for a given visit using the `x-search_engine_name` field.
Search Page Ranking	numeric	Represents the result order of the site listed by the search engine. It is the field where the `x-visit_search_engine_name` is not empty.
Visit Number	simple	Shows the number of the visit for a specific visitor. It is the first non-blank value for a given visit using the `x-visit_num` field.
Referrer Type	simple	Identifies the referrer based on a specified type. It is the first row value for a given visit using the x-ref_type_lookup field when the x-ref_domain is not empty.
Paid Search Flag	simple	Provides a marker to identify if the searches came from a paid source. It is the field when `x-visit_search_engine_name` is not empty.
Order	numeric	Identifies hits where an order occurred. Conditional when `x-purchaseid` is not empty. The parent of Order is Hit.
Entry Page	simple	Represents the first page of each visit. It is the first non-blank value for a given visit using the `x-pagename` field.
Pages Path Length	numeric	Counts the number of pages. The dimension is a COUNT operation using the `x-visit_page_num` field.
Site Section Path Length	numeric	The dimension is a COUNT operation using the `x-channel` field

Dimension Name	Dimension Type	Description
Hours of Today Last 24 Hours Last 48 Hours	calculated report	Reports visits based on the number of identified hours. These calculated report dimensions use the `Hour` dimension and the `Report Time` metric.
Days ago Last 21 days Last 7 days Today Yesterday	calculated report	Reports visits based on the number of identified days. These calculated report dimensions use the `Day` dimension and the `Report Time` metric.
Last 4 Weeks Last 8 Weeks Last Week This Week Weeks Ago	calculated report	Reports visits based on the number of identified weeks. These calculated report dimensions use the `Week` dimension and the `Report Time` metric.
Last 3 Months Last Month Months Ago This Month	calculated report	Reports visits based on the number of identified months. These calculated report dimensions use the `Month` dimension and the `Report Time` metric.
Visit Duration	metric	Summarizes the exact page duration for each visit by multiplying the values by 0 .1. The dimension subsumes the times into 60 second chunks (1 minute).
Time Spent per Visit	prefix	Categorizes the duration of visits using the `Visit Duration` dimension. This dimension groups the visit duration values based on the length of the visits.

Visitor Dimensions

Visitor-based dimensions capture data about visitor characteristics. Subordinate dimensions listed below are built from the parent `Visitor` dimension.

Dimension Name	Dimension Type	Description
Visitor	countable	Counts all visitors in the system, keying on the `x-trackingid` field. The `Visitor` dimension is the root dimension.
IP	simple	Lists IP addresses for each visitor. If changed to a `Denormal` dimension type, then the `IP` dimension can be used to isolate specific visitors or investigate visitor splitting. This dimension uses the `c-ip` field.

Dimension Name	Dimension Type	Description
Visitor State	simple	Identifies the geographical region of the visitor. This data is passed in using javascript code on a selected page. This dimension is usually set on a purchase page. It is the first non-blank value for a given visitor using the `x-state` field.
Visitor Zip	simple	Identifies the zip code of the visitor. The zip code is passed in using javascript code on a selected page. This dimension is usually set on a purchase page. It is the first non-blank value for a given visitor using the `x-zip` field.
Browser	simple	Lists browsers and their versions from each visitor. It is the first non-blank value for a given visitor using the `x-browser` field.
Browser Type	simple	Identifies the browser type without any additional version information. The dimension allows a maximum of 10 elements. It is the first non-blank value for a given visitor using the `x-browser-type` field.
Browser Height	simple	Shows the height of the browser. It is the first non-blank value for a visitor using the `x-browser_height` field. .
Browser Heights Browser Screen Height	metric	Groups browser heights similar to groups established in SiteCatalyst for integrated reporting.
Browser Width	simple	Displays browser widths of visitors to the site. It is the first non-blank value for a visitor using the `x-browser_width` field.
Browser Widths Browser Screen Width	metric	Groups browser widths similar to groups established in Site Catalyst for integrated reporting.
Color	simple	Shows different color depths for each visitor. It is the first non-blank value for a given visitor using the `x-c_color` field.
Monitor Color Depth	metric	Identifies the color depth settings for computer monitors using the `x-color-depth` dimension.
Client Connection Type	simple	Identifies the method each visitor uses to connect to the Internet. It is the first non-blank value for a given visitor using the `x-ct_connection_type` field.
Server Connection Type	simple	Identifies the method used to connect to servers. It is the first non-blank value for a given visitor using the `x-connection_type_name` field.
Visitor Country	simple	Identifies the country from which the visitor is connecting. It is the first non-blank value for a given visitor using the `x-country_lookup` field. The Visitor Country dimension is passed in using javascript code on the selected page, usually a purchase page.
Geo US State	simple	Populates only when the `x-geo_country` dimension matches the value `usa`. It is the first row for a given visitor using the `x-geo_region` field.
Geo Cities	simple	Represents the city from which the visitor traffic originated. It is the first non-blank row for a given visitor using the `x-geo_city` field.
Geo DMA	simple	Represents the DMA from which the visitor traffic originated. It is the first non-blank row for a given visitor using the `x-geo_dma` field.

Dimension Name	Dimension Type	Description
Geo US DMA	simple	Populates only when the `x-geo_country` field matches the value `usa`. It is the first row for a given visitor using the `x-geo_dma` field.
Geo Regions	simple	Identifies the geographical region of a visitor. It is the first non-blank value for a given visitor using the `x-geo_region` field.
Java Enabled	simple	Flags visitors if java is enabled in their environment. It is the first non-blank value for a given visitor using the `x-java_enabled` field.
Javascript Version	simple	Identifies the version of Javascript that each visitor is using. It is the first non-blank value for a given visitor using the `x-javascript_lookup` field.
Language	simple	Identifies the most likely language of the visitor. It is the first non-blank value for a given visitor using the `x-languages_lookup` field.
Operating System	simple	Identifies the operating system of the visitor. It is the first non-blank value for a given visitor using the `x-os_lookup` table.
Third Party Cookie	simple	Allows for visitors to be identified based on those with or without persistent cookies. It is the first non-blank value for a given visitor using the `x-persistent_cookie` field.
Monitor Resolution	simple	Identifies the monitor resolution for each visitor. It is the first non-blank value for a given visitor using the `x-resolution_lookup` field.
Domains	simple	Identifies the network domain for the visitor's internet service provider (ISP). It is the first non-blank value for a given visitor using the `x-domain field`.

Metrics

Description of Metrics and their equations for Adobe Site Catalyst to Adobe Insight data feeds.

Metric	Equation	Description
Visitors	sum(one, visitor)	Sum total of all visitors collected in the data set.
Visits	sum(one, visit)[Third_Party_Cookie=”y”]	Sum total of all visits in the data set when a third-party cookie is present.
Hits Instances	sum(one,Hit)	Hits and Instances are the sum total of all hits in the data set.
Page Views	Hits[Page_event_type=”0”]	Sum of all hits in the data set where a page view was marked.
Page Views No Reloads	Page_Views [reload_flag<>"Reload"]	Page views that are not marked as a `reloaded page`.
Reloads	Page_Views[reload_flag="Reload"]	Page views marked as `reload` using the reload flag.
Avg Visitor Visits	(Visits by Visitor) / Visitors	Average number of visits per visitor across the data set.

Metric	Equation	Description
Entries	Hits[no shift(None,Hit,Visit,-1)]	Count of entries to the site as calculated by hits where no previous hit was found for the visit.
Exits	Hits[no shift(None,Hit,Visit,1)]	Count of exits from the site as calculated by hits where no further hit was found for the visit.
Exit Rate	Exits/Hits	Number of exits divided by the number of hits in the data set.
Hits by Visit	Hits by Visit	Hits by Visit is the number of hits using the “by” operand on Visit.
Hits per Visit	Hits/(Visits by Hit)	Calculation using the total number of hits divided by “Visits by Hit”.
Pct of Visitors	Visitors/total(Visitors)	Percentage of visitors of the element divided by the total visitors of the data set.
Pct of Visits	Visits/total(Visits)	Percentage of visits of the element divided by the total visits of the data set.
Pct of Hits	Hits/total(Hits)	Percentage of hits of the element divided by the total hits of the data set.
Raw Visitors	raw(Visitors)	The raw count of Visitors in a data set. This numbers is usually the same as the number of Visitors.
Raw Visits	raw(Visits)	Raw count of visits in the data set. This number is usually the same as Visits.
Raw Hits	raw(Hits)	Raw count of hits in the data set. This is usually the same as Hits.
Visitors by Visit	Visitors by Visit	Number of visitors using the “by” operand on Visit.
Return Visits	Visits[Visit_Number > "1"]	Number of visits that have a Visit Number value greater than 1.
Retention	Visits[shift(None,Visit,Visitor,1) = ""] / Visits	Percentage of visits that are not the visitors’ last visit.
Visit Duration	(sum(Exact_Page_Duration,Visit)*.1/Visits) [Exact_Page_Duration <= '01:00:00']	Total of page views with a duration less than or equal to one hour.
Page View Duration	sum(exact_page_duration,Hit)*0.1/hits[any Exact_Page_Duration]	Length of time between page views.
LVCI90	(raw(Visitors) - ((raw(Visitors) + .69)^0.5 * 1.281551 - 1.2269))*(Visitors/raw(Visitors))	Measure of the lowest number of possible visitors as reported by Adobe Insight. Mathematically, it specifies the lowest number of visitors with a 90 percent probability.
UVCI90	(((raw(Visitors) + .68)^0.5 * 1.281551 + 1.2269) + raw(Visitors))*(Visitors/raw(Visitors))	Measure of the highest number of possible visitors as reported by Adobe Insight. Mathematically, it specifies the highest

Metric	Equation	Description
		number of visitors with a 90 percent probability.
VCI80	((raw(Visitors) + .68)^0.5 * 1.281551 + 1.2269) / raw(Visitors)	Measure of confidence of the Visitors metric as reported by Adobe Insight. Mathematically it is a +/- percentage specifying the range within which the actual answer will lie 80 percent of the time. Traditionally, doubling the VCI80 percentage will give a range within which the actual answer will lie 90 percent of the time.
Products	sum(One, Product)	Total of products in the data set.
Revenue	sum(Revenue, Product)	Total revenue for a given product.
Visit Revenue	sum(Revenue, Visit)	Total revenue for each visit.
Orders	sum(Order, Hit)	Total orders for a given product.
Visit Orders	sum(Order, Visit)	Total orders for each visit.
Units	sum(Unit, Product)	Total units for a given product.
Visit Units	sum(Unit,Visit)	Total units per visit.
Carts	Hits[Carts="1"]	Number of hits where a cart was present.
Cart Views	Hits[Cart_Views="1"]	Number of hits where a cart view was present.
Cart Additions	Hits[Cart_Additions="1"]	Number of hits where a cart addition was present.
Cart Removals	Hits[Cart_Removals="1"]	Number of hits where a cart removal occurred.
Checkouts	Hits[Checkouts="1"]	Number of hits where a checkout occurred.
Clickthroughs	sum(one, Clickthrough)	Sum total of all the click-throughs in the data set
Referrer Clickthroughs	Clickthroughs [Referring_Domain_Clickthrough<>"None" and (Page_Event_Type_Clickthrough="0" or Page_Event_Type_Clickthrough="100" or Page_Event_Type_Clickthrough="101" or Page_Event_Type_Clickthrough="102" or Page_Event_Type_Clickthrough="76")]	Number of click-throughs that exist for a populated Referring Domain Clickthrough and exist on initiation of Page Event Types.
Referrer Instances	Hits [Page_Event_Type="0" or Page_Event_Type="100" or Page_Event_Type="101" or Page_Event_Type="102" or Page_Event_Type="76"]	Number of hits that exist for initiation of Page Event Types.
Purchased Products	Products [Order="1"]	Number of products associated with an order.

Metric	Equation	Description
Viewed Products	Products [Viewed_Products <> "None"]	Number of products associated with a view interaction.
Custom Events Event #	sum(cust#, Hit)	Sum of all customer specified custom events.