|  |  |
| --- | --- |
|  | **Unidev** |
|  | **Unified Development, Inc.**  16690 Swingley Ridge Road  Suite 260  Chesterfield, MO 63017  (636) 532-4424 |

|  |
| --- |
|  |
|  |

**Duncan Solutions, Inc**.

Web Services

April 2014

Revision 1.00

**Prepared by:**

Caleb Miller

Ron Howard

Unified Development

**Table of Contents**

[1. Overview 4](#_Toc384800006)

[2. Target Audience 5](#_Toc384800007)

[3. Enforcement Database (EDB) 6](#_Toc384800008)

[4. Table Definitions 8](#_Toc384800009)

[5. Citations 9](#_Toc384800010)

[5.1 Citation 10](#_Toc384800011)

[5.2 Citation Type 11](#_Toc384800012)

[5.3 Citation Version 12](#_Toc384800013)

[5.4 Citation Definition 13](#_Toc384800014)

[5.5 Citation Field Definition 14](#_Toc384800015)

[5.6 Data Type 15](#_Toc384800016)

[5.7 Citation Field Value 16](#_Toc384800017)

[6. Citation Groups 18](#_Toc384800018)

[6.1 Citation Group 20](#_Toc384800019)

[6.2 Citation Group Field 21](#_Toc384800020)

[6.3 Field Format 22](#_Toc384800021)

[7. Citation Filters 23](#_Toc384800022)

[7.1 Citation Filter Group 24](#_Toc384800023)

[7.2 Citation Filter Type 26](#_Toc384800024)

[7.3 Citation Filter Value 27](#_Toc384800025)

[8. Lists 28](#_Toc384800026)

[8.1 List 29](#_Toc384800027)

[8.2 List Definition 30](#_Toc384800028)

[8.3 List Version 31](#_Toc384800029)

[8.4 List Field Definition 32](#_Toc384800030)

[8.5 List Field Value 33](#_Toc384800031)

[9. Assumptions 35](#_Toc384800032)

**Revision** History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Author | Description | Revision Date |
| 1.00 | Caleb Miller | Initial version | 04/9/2014 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Overview

The purpose of this document is to explain the architecture proposed for the Enforcement database (EDB) that will be used with the Ticketing Handheld device (HH) and the integration with the Auto Issue application (AI). It will include expected business rules, assumptions and logic flows on how to sync data back and forth from the AI system.

This document will cover the following areas:

1. Database Architecture and Design
   1. Table Definitions
      1. Column Definitions
   2. Table Relations
   3. Suggested Indexes
   4. Immutable values
2. Data Administration
   1. Screens and rules required to administer the data in the system
   2. Merging data to and from the AI system
   3. Versioning data
   4. Defining display rules
3. Displaying Data to the user
   1. Assumed business rules
   2. Dynamic Index page filters
      1. Forcing Required fields
      2. Different Field Types
      3. Displaying data as filters
      4. Formatting data
   3. Dynamic Grid Columns
      1. Displaying data in the grid
      2. Formatting Data
   4. Citation Detail Pages
      1. Grouping Fields
      2. Formatting fields

# Target Audience

The target audience of this document is person or persons who have:

1. Experience in the following technologies:
   1. C# /.Net /
   2. T-SQL, SSMS, SqlServer, Stored Procedures and Views, Indexes
2. The user of this document has a full understanding of the Duncan PEMS project. This includes database and system architecture knowledge, desired business rules of the application, etc. Most importantly this includes the handheld device, Auto Issue, and how that data is generated and managed. This document will not elaborate on those items. The user must know how data comes and goes from the AI system, the format of the data, etc.
3. Microsoft SQL Server administration and understand rights, database creation and administration, and are able to use either SSMS or SQL command line interface.
4. Visual Studio 2012

# Enforcement Database (EDB)

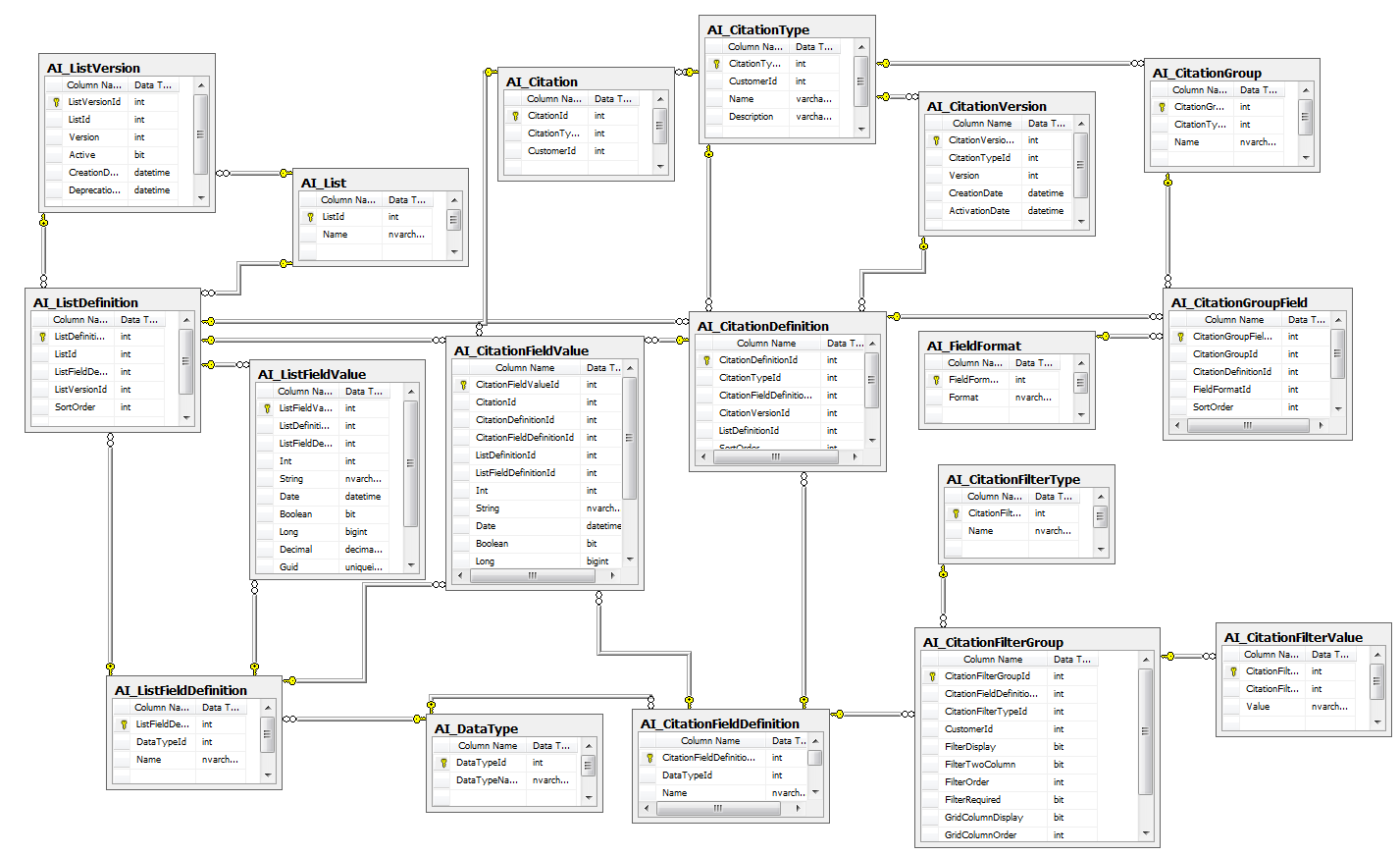
Apart from having to be a system that supports multiple customers with a dynamic data structure, the EBD has two main goals:

1. Manage the various citations and citation types for each customer. This system should allow each customer to define multiple types of citations, the fields that the citations are comprised of, versioning of the citation, etc.
2. Manage the list definitions for some of the available data for those citations. A list of states and all of the states associated data is a good example of this, or a list of possible citations (ticket ids) for a particular type of ticket.
3. NOTE: The current schema does not have indexes applied to it, as discussion and review is needed before we get to this step. Further analysis will determine the Indexes to be applied to the EDB.
4. Since this data is sensitive to the time it was inserted, most data will not be editable within the system. Once a list is created, it has to stay the same forever, same with citation types, versions, citation definitions, etc. This is for auditing purposes in the future and bringing up past citations for a specific time / version / customer, etc.

Install Script: TODO – re-generate this once the final solution is determined this to make sure it is up to date. Same with main db diagram and the 4 sub diagrams



Complete Data Diagram:



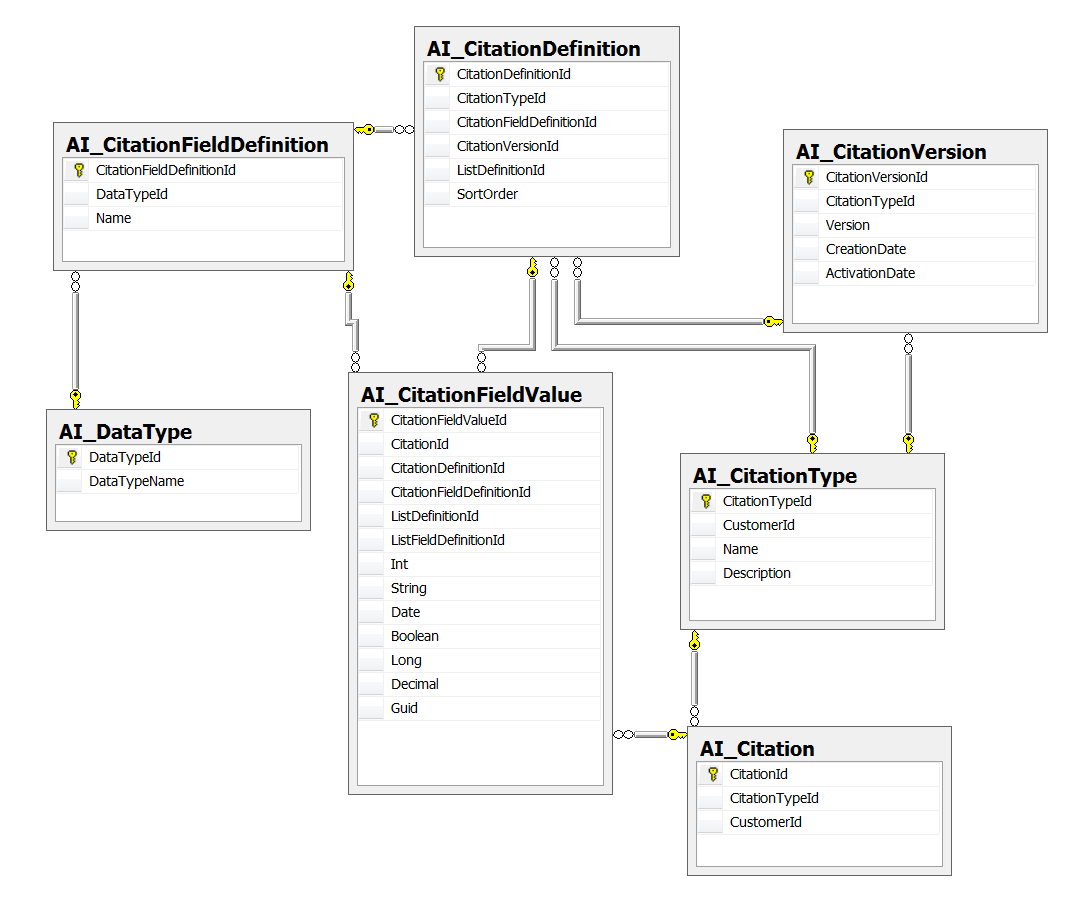
# Table Definitions

This section will list each table along with a summary of the data represented, any assumptions that were made in the process of determining the table structure, business rules for loading the data, and what each Column represents. The Primary Keys (PK) and Foreign keys (FK) will also be called out.

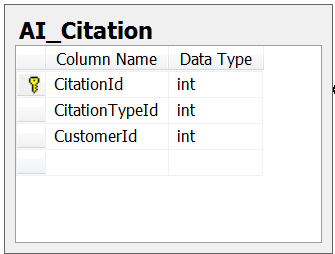
This section will cover the following concepts

1. Citations
   1. Customer specific, versioned citation types and all of the associated data
2. Citation Groups
   1. Grouping of citation fields to display to the user on the citation details page
3. Citation Filters
   1. Grouping of citation fields to display to the user in the filter section in the index page for citations as well as defining the columns displayed on a per customer basis.
4. Lists
   1. Customer specific, versioned lists that will be used in conjunction with citations to determine possible citation values on a version by version basis.

# Citations



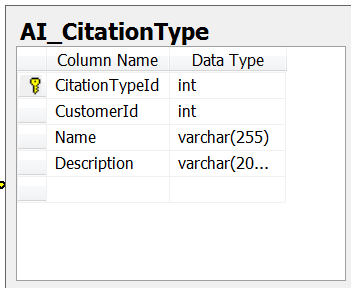
## Citation



This data represents an individual citation for a given customer.

1. Citation Id: PK, Auto Increment.
   1. This is the PK for the table and will be used throughout the system to refer to a specific instance of a citation (Ex: Parking Ticket) for a customer.
2. Citation Type Id: FK to the Citation Types
   1. Defines the type of citation (Parking, Storefront, etc)
3. Customer Id: the customer id from the customer profile table

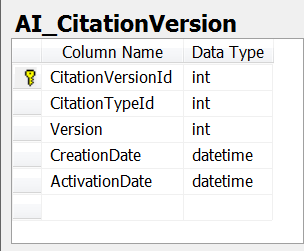
## Citation Type



This Data represents a type of citation for a customer

1. Citation Type Id: PK, Auto Increment
2. Customer Id: the customer id from the customer profile table
3. Name: the name of the Citation type. . An example would be Parking Tickets, Store front tickets, Moving Violation, etc.
4. Description: Short description of the citation type. This is intended to be a more user friendly description of the citation type, as the name might be an aggregated value of some sort that is not very human readable (PK29-525A for example).

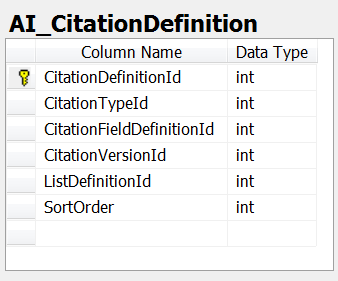
## Citation Version



This table allows the system to version citation types. The application will add a version when the definition of a citation changes. This will allow the application to retrieve information on a specific version of a citation without having to store multiple versions in the Citation Type table.

1. Citation Version Id: PK, Auto Increment
2. Citation Type Id: FK to the Citation Type
3. Version: the version of the citation type. This is incremented each time the definition of a citation is updated (Citation Definition).
4. Creation Date: The date that the version of the Citation was created
5. Activation Date: The intended activation date for the citation.
   1. This allows the administrators to create a citation definition and set the publish date in the future.
   2. Example: if they know a parking ticket is going to change in a month, then they can create a new definition of the parking ticket and set the date of activation to when it should be pushed to the handheld.
   3. The assumption is the handheld will be receiving the most recent, activated version of the citation.

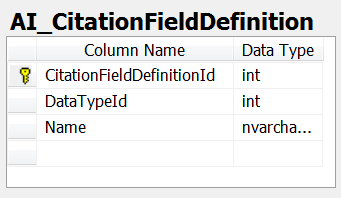
## Citation Definition



This data is the definition of all the fields that are associated with a specific version of a citation. If we have a parking citation with two fields, then this table will include two rows, one for each field.

1. Citation Definition Id: PK, auto increment
2. Citation Type Id: the type of citation
3. Citation Field Definition Id: the citation field that is associated with this definition
4. Citation Version Id: the version of this definition. This will allow us to get the definition of a citation for any version of any citation type in the system, current or historical.
5. List Definition Id: Nullable field
   1. This is an optional field that will relate the field on the citation to a list of items. So if this definition row is referencing the “State” Citation Field Definition, it might also use the “States” list definition.
   2. This allows us to get a specific versioned list of items that were available when the ticket was created. It allows us to bring back the original list of items for that field on specific version of the ticket, when the ticket was being issued. This will let us re-generate the ticket form exactly as it was when the ticket was being created, regardless of when it was created.
6. Sort Order: Ordering of the citation definition. When the data is pushed to the handheld, it will respect this order and display the data accordingly. This will let the administrators define the order of the fields that are displayed for each ticket version for each customer.

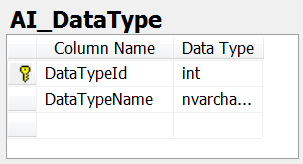
## Citation Field Definition



This data represents the unique list of all the possible fields for all tickets in the system. The Type / Name will be a virtual (business logic) composite key, so no two rows will have the same name AND Data type assigned to it. There can only be one “Officer Id” that is an Int value, etc. If there are 854 different fields on all the citations for all the customers in the system, this table will have 854 rows.

1. Citation Field Definition Id: PK, auto increment
2. Data Type Id: The data type for this field definition. Example: Int, String, Datetime, etc.
3. Name: The name of the field that the data represents. Example: “State”, “LicNumber”, etc.

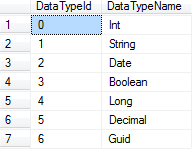
## Data Type



This table represents the possible data types in the system for field definitions (includes lists and citations). The data in this table directly corrosponds to the columns in the Citation Field Values and List Field Values columns and is used to determine where to store / retrieve the data from. The idea is to eliminate the need to do string conversions to determine the values of the data in the system. If the application has to convert all the saved values from a string to the appropriate data type, performance will be affected, especially considering the volumne of data that will be stored in these specific tables (they are the most data heavy in the EDB).

1. Data Type Id: PK, NOT auto Increment
2. Data Type Name: Fiendly name of the Sql DB Type.

Here are the values that will be in this table:



And they directly represent the following columns:

Citation Field Value

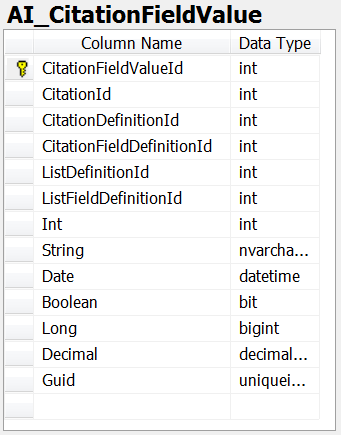


List Field Value



When storing data, the type will be looked up and the data will be inserted in the appropriate column. The same process will be followed for retrieval of data.

## Citation Field Value



This table represents all of the citation field values for all customers, citation versions, etc in the system. When an enforcement officer writes a new citation, it is broken down and inserted into this table based on the Citation Definition for that version of the Citation Type.

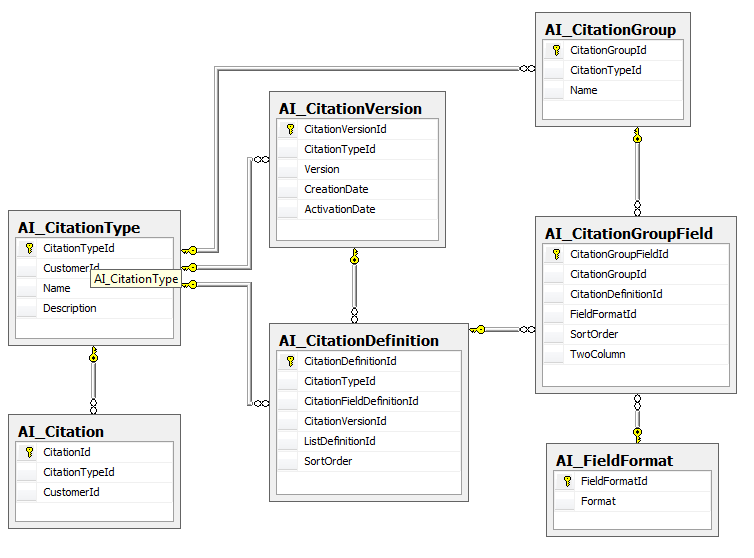
NOTE: This table will need to be indexed; more review is needed before we can determine the exact indexes.

TODO : add indexes

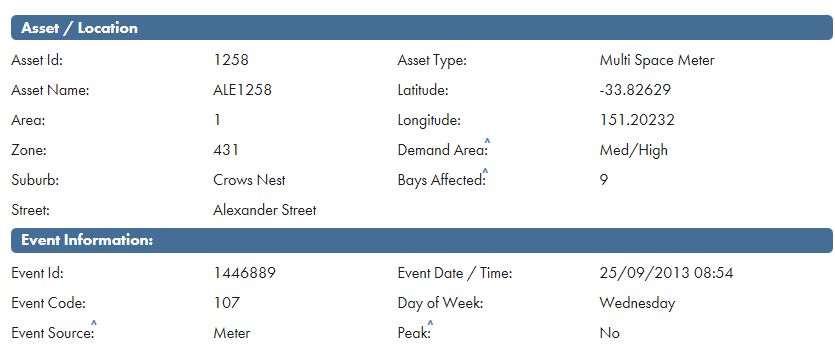
TODO: might need to add timestamp as well so we can determine when the values were inserted. This will help us get specific list versions for a ticket. Might also add this to the citation itself, or both.

1. Citation Field Value Id: PK, auto increment
2. Citation Id: The id of the citation that these values are being stored for.
3. Citation Definition Id: The definition of the citation this data is for. This allows us to get the entire citation off of any field value in the citation.
4. Citation Field Definition Id: The field that this value represents. Important to be able to determine the type of data being stored, the version of the citation type, etc.
5. List Definition Id: The version of the list that this field represented at the time of data insertion. This allows us to get the specific version of the list and all of its data elements within that list.
   1. This is a nullable field and is only used for citation fields that represent list data. Example: a list of “States”.
6. List Field Definition Id: The id of the specific field associated with this list definition. So if the list definition for this row is States, then this field lets us know that the data being stored came from the “State Motto” or “State Name” field of that list. This is a nullable field and is not required.
7. Data type columns
   1. Each column represents a different type of data. The value inserted for each row in this table will only exist in one of these columns, based on the type of data being inserted (inferred from the Field Definition FK).
   2. Example: State of “Missouri”, then the data will be inserted into the “String” column and all of the others will be left null. Officer ID of 42 will be inserted into the “Int” field.

# Citation Groups



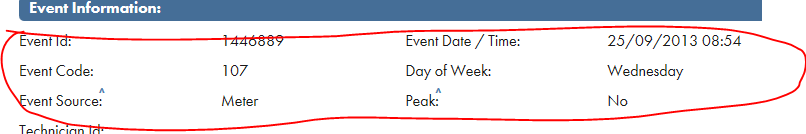
The purpose of these tables is to define the groups and fields that are displayed within those groups on a citation details page. Similar to this example of the Event Details Page:



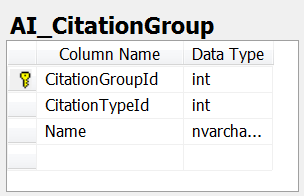
These tables will allow the administrators to define the groups for a citation type:

as well as the fields associated with that group, the order of the fields, the format of the fields, and which fields to display:



## Citation Group



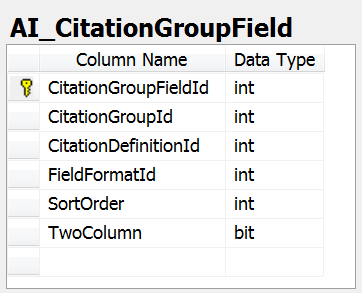
This table represents the groupings that will be displayed on the details page for a specific Citation Type. The administrator will define the groups for a citation details page. In the example the event detail has two sections “Asset / Location” and Event Information.

TODO : We might have to add a Citation Version ID as a FKto this as well, so we can define different fields for different versions of citations.

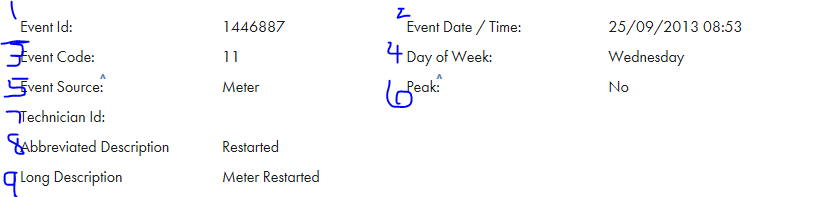
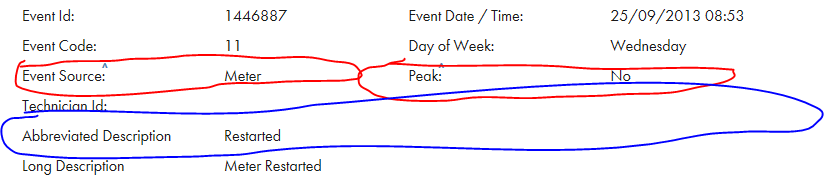


1. Citation Group Id: PK, auto increment
2. Citation Type Id: The type of citation that this group will be associated with.
3. Name: the name of the group. Example “ Asset / Location”

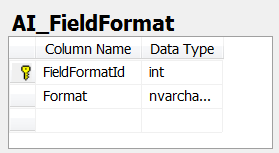
## Citation Group Field



This table represents the fields that will be part of a citation group, the order of those fields, and if they are to be displayed on one column or two.

1. Citation Group Field Id: PK, auto increment
2. Citation Group Id: the group that this field is associated with
3. Citation definition id: the citation definition so the application can determine the specific version this data is referring to
4. Field Format Id: the format of this field (“dd/mm/yyyy” or HH:MM, etc). This is a nullable field
5. Sort Order: the order in which the data should be displayed on the screen, floating from left to right like so:
   1. 
6. Two Column: should we display data on a single line or in the two column format
   1. In this example, a one column is circled in blue and the two columns is circled in red
   2. 

## Field Format

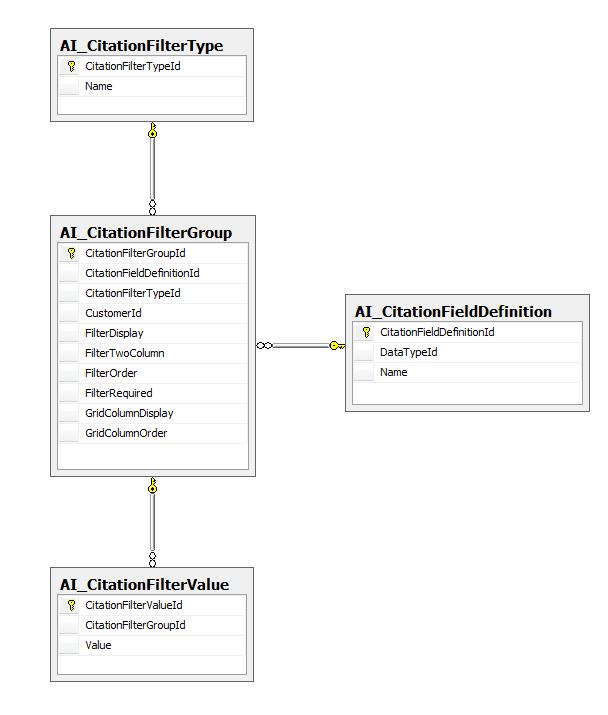


This table represents all of the possible formats that can be applied to fields in the system. This will be a hard coded list of values that will be added as needed. Examples include “HH:MM”, “ToLocalTime”, etc.

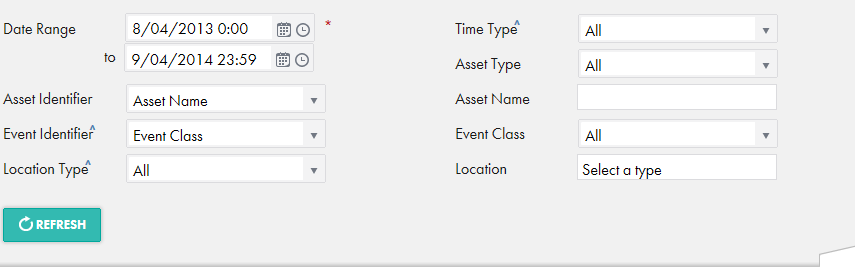
We will ony be able to add to this table, never remove or edit.

1. Field Format Id: PK, auto increment
2. Format: the string format to be applied.
   1. In .Net / c#, assuming fieldFormat is the value of this column, we will apply like SomeValue.ToString(fieldFormat);

# Citation Filters



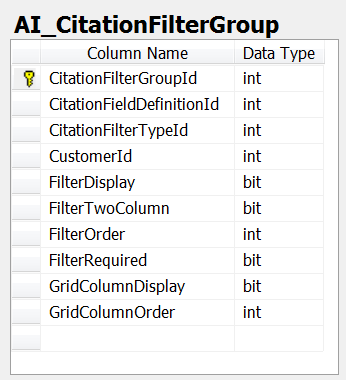
The purpose of these tables is to allow the administrators to define the filters that are displayed on the citations index page (event index page example):



And the grids columns to show:

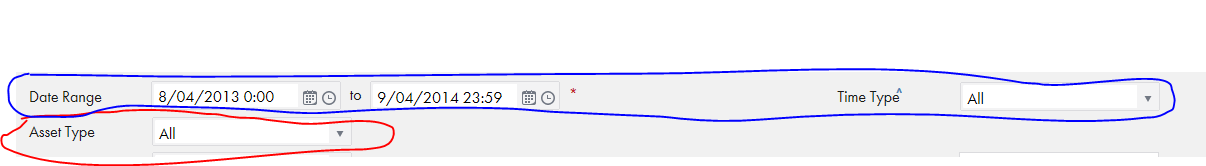


## Citation Filter Group

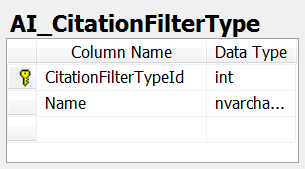


This table represents configuration of the citation index page. Administrators will be able to specify what filters to display, how to display them (date times, drop down lists, etc,) and if the application should display them within the grid.

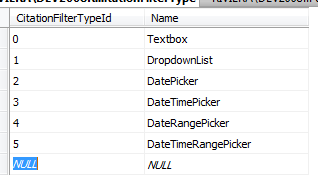
The assumption is that as field values are being inserted into the Citation Field Value table, they are also being check for a unique field definition here and added if needed. Administration screen for this will be able to modify the data in this table (all non-FK fields) but not add so it.

1. Citation Filter Group Id : PK, auto increment
2. Citation Field Definition Id: the field definition for this filter. When setting this up, the application will need to display a list of field definitions that are in the system for that customer so they can choose it to be a filter.
3. Citation Filter Type Id: The type of filter they wish to display (dropdown list, date range picker, etc). This should default to the first value in the citation filter type table, which is “TextBox”
4. Customer Id: the customer this filter is assigned to
5. Filter Display: Boolean value to represent if the application should display this field definition as a filter in the filters section
6. Filter Two Column: determines if the filter should be on its own line or in the two column format (blue is two column, red is one column in the example below)
   1. 
7. Filter Order: sort order of the filter to display, from left to right floating. In the above example it would be
   1. DateRange: 0
   2. TimeType: 1
   3. AssetType: 2
8. Filter Required: denotes if the filter is a required filter or not
   1. Example: 
9. Grid Column Display: denotes if the application should display the field as a grid column
10. Grid Column Order: the order of the column in the grid, from left to right.

## Citation Filter Type



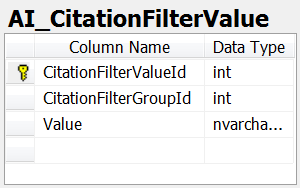
This table represents the types of filters in the system. This will be an immutable list, and the data will be as follows:



When rendering the filters, a switch statement should be used to determine the html to output.

1. Citation Filter Type Id: PK, NOT auto increment
2. Name: Name of the filter type.

## Citation Filter Value



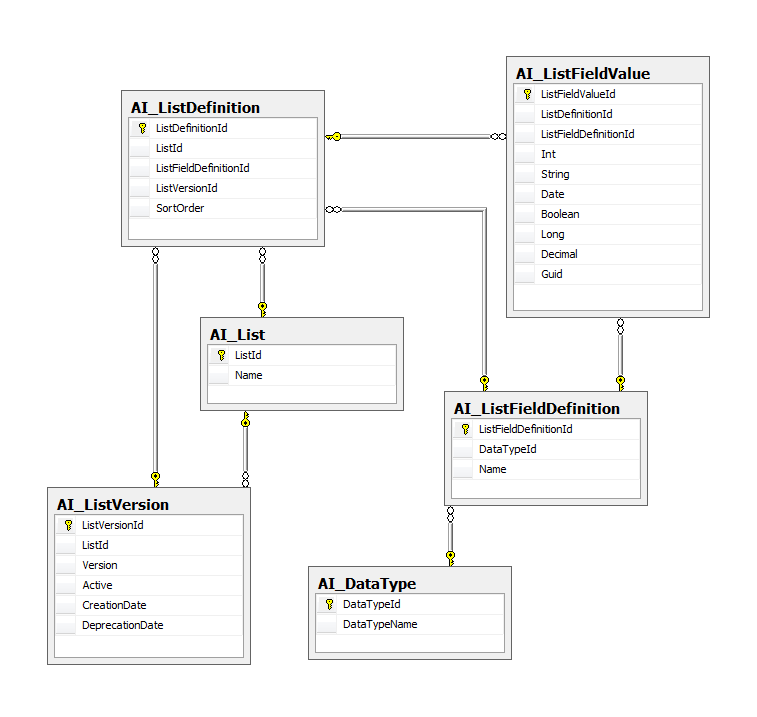
This table represents the possible data that will be displayed when the filter type is set to a Drop Down list. For Example, if the field definition is “States”, then this list will be comprised of the list of states that will be displayed in the filter on the index page. This table is only used then the administrator wishes to display a filter as a Dropdown list.

This will allow administrators to add additional list values if they haven’t been inserted into the system yet. This list does not directly represent the items for the List Definition, as those are specific to a version of a list.

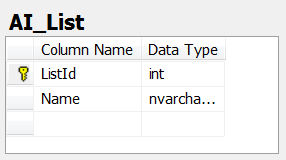
The items in this list are added during the data insert process (add if doesn’t exist), but cannot be modified. They can be removed from this if the administrators determine they do not want a list item to be displayed for a filter value.

1. Citation Filter Value ID: PK, auto increment
2. Citation Filter Group Id: the filter group this value is associated with
3. Value: The value to display in the drop down list

# Lists



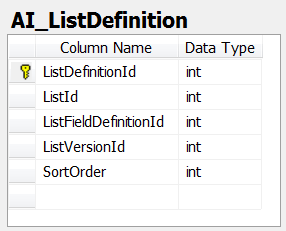
## List



This table represents a list of data within the system. An example of this would be a list of States with the fields of “State Abbreviation”, “State Name”, and “State Motto”.

1. List Id: PK, auto increment
2. Name: the name of the list. Example “States”

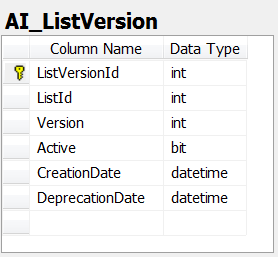
## List Definition



This table represents all of the fields that are comprised of a list for a specific version. For example version 1 of the “States” list might have Name and Abbreviation, whereas version 2 might have Name, Abbreviation, and Motto.

1. List Definition Id: PK, auto increment
2. List Id: The associated list for this definition
3. List Field Definition Id: the field for this definition row. IE: State Name
4. List Version Id: the associated version for this list definition. This will allow us to get a list of fields and values for a specific version of a list.
5. Sort Order: the order of the field in the list. When pushing data to the handheld, we will sort by this value before passing it on.

## List Version



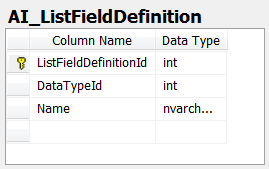
This table represents a specific version of a list.

1. List Version Id: PK, auto increment
2. List Id: The associated list for this version
3. Version: The version of the list
4. Active: Determines whether the list is active or not. If it is active it cannot be changed. This flag is really set to determine if the list is still being edited. Once the administrator is done configuring a list, then the list is saved, the active flag is set to true, and this version of the list cannot be modified any more.
5. Creation Date: The date that the list was first created
6. Deprecation Date: The date a new version of the same list was created.

Since we do not have the version of a list when inserting a new ticket (the current AI host system doesn’t version lists), in order to determine the list definition to include in the Citation Field Value table, we will need to take the time that the citation as created (The timestamp on the citation either in field values or the citation itself, TBD - TODO) and do a comparison to the List version creation date and deprecation date where the citation creation date is between the list version creation and deprecation date.

This will allow us to determine the exact set of lists that were available when the citation was being generated. If we need to generate a ticket from 6 months ago, we will be able to determine the lists used on that ticket to exactly replicate the ticket as it was 6 months ago using the same set of list data as well as field data.

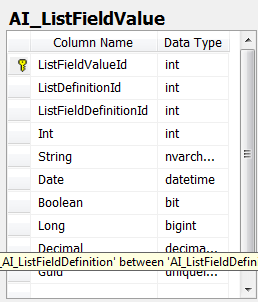
## List Field Definition



This data represents the unique list of all the possible fields for all lists in the system. The Type / Name will be a virtual (business logic) composite key, so no two rows will have the same name AND Data type assigned to it. There can only be one “State Motto” that is a String value, etc. If there are 854 different fields on all the Lists for all the customers in the system, this table will have 854 rows.

1. List Field Definition Id: PK, auto increment
2. Data Type Id: The data type for this field definition. Example: Int, String, Datetime, etc.
3. Name: The name of the field that the data represents. Example: “State Name”, “State Motto”, etc.

## List Field Value



This table represents all of the list field values for all customers, list versions, etc in the system. When a list is created, it is broken down and inserted into this table based on the List Definition for that version of the List Type.

NOTE: This table will need to be indexed; more review is needed before we can determine the exact indexes.

TODO : add indexes – timestamp at least (possibly)

TODO: might need to add timestamp as well so we can determine when the values were inserted. This will help us get specific list versions for a ticket.

1. List Field Value Id: PK, auto increment
2. List Definition Id: The version of the list that this field represented at the time of data insertion. This allows us to get the specific version of the list and all of its data elements within that list.
   1. This is a nullable field and is only used for citation fields that represent list data. Example: a list of “States”.
3. List Field Definition Id: The id of the specific field associated with this list definition. So if the list definition for this row is States, then this field lets us know that the data being stored came from the “State Motto” or “State Name” field of that list. This is a nullable field and is not required.
4. Data type columns
   1. Each column represents a different type of data. The value inserted for each row in this table will only exist in one of these columns, based on the type of data being inserted (inferred from the Field Definition FK).
   2. Example: State Name of “Missouri”, then the data will be inserted into the “String” column and all of the others will be left null. Officer ID of 42 will be inserted into the “Int” field.

Todo – citation group and field def – versions for this. Might have to add a citation version, and each version of a ticket could have a different set of groups / fields per group.

# Assumptions

1. On the filters page there will be some hard coded filters applied to all index pages for all customers. The reason for this is because these filters will not be coming from the possible Citation field value data, but citation data itself
   1. Citation Type
   2. Citation Version
   3. Citation Creation Date – if we add timestamp to the citation (which we should I think)\_-TODO