MaxMind Forum

Welcome to the MaxMind Community

Skip to content

Search… Search
Advanced search

- Board index ‹ Geolocation (Public Section) ‹ Open Source Projects with GeoLite Support
- Change font size

- FAQ
- Register
- Login

Fast SQL Queries Using the CSV Database

Post a reply

Search this topic… Search

2 posts • Page **1** of **1**

Fast SQL Queries Using the CSV Database

by **Bigal21** » Wed Aug 12, 2009 9:19 pm

There is a way to use the CSV file with lightening quick access times using SQL. The hard part is getting the Location ID (within an IP address range)... due to the nature of binary searches. Once you have the Location ID, you can get the City/Country details quickly (from another table) if you index the Location ID column.

The first step is importing IP blocks and location ID's into a table. It is important to make the "Start IP" column the primary key. You also need a Counter column which is incremented sequentially (counter++) during the import process. You must put an index on the counter column. Here is a table definition.

```
CREATE TABLE IF NOT EXISTS maxmindlocationids (
StartIP int(11) UNSIGNED,
EndIP int(11) UNSIGNED,
Counter int(11),
LocationID int(8),
PRIMARY KEY (StartIP),

INDEX maxmindlocationids_Counter (Counter)
```

```
);
```

During the import process... make sure to check the CSV file for errors (although I didn't find any). The CSV file should have the StartIP ascending with no duplicate values.

It will take two queries to get the Location ID. Below is some PHP code outlining the algorithm.

```
// Will return NULL if the IP address does come across a match.
static function getLocationIdFromIPaddress($ipAddress){

self::validateIPaddress($ipAddress);

$dbCmd = new DbCmd();

// Get the next highest record (above the row we need).
$dbCmd->Query("SELECT Counter FROM maxmindlocationids WHERE StartIP >
".self::getNumericIP($ipAddress)." ORDER BY StartIP ASC LIMIT 1");
if($dbCmd->GetNumRows() == 0)
return null;

$couterLocation = $dbCmd->GetValue();

// We know the record we are looking for is possibly one record lower.
$couterLocation--;

// In case we are on the first row.
if($couterLocation == 0)
return null;

$dbCmd->Query("SELECT LocationID, EndIP FROM maxmindlocationids WHERE
Counter=$couterLocation");
if($dbCmd->GetNumRows() == 0)
return null;

$row = $dbCmd->GetRow();

// We need to make sure that there is not a "hole" between IP ranges. So we have to check the END IP of
this record.
if($row["EndIP"] < self::getNumericIP($ipAddress))
return null;

return $row["LocationID"];
}
```

Bigal21

Posts: 1
Joined: Wed Aug 12, 2009 8:59 pm

Top

---

by **RadiusKuntoro** » Tue Apr 06, 2010 5:47 pm

Thanks for the tutorial.

Instead of doing 2 queries, I find it easier to do a query with LIMIT 2.

RadiusKuntoro

Posts: 1
Joined: Tue Apr 06, 2010 5:27 pm

Top

Display posts from previous: All posts Sort by Post time Ascending Go

---

Post a reply
2 posts • Page **1** of **1**

Return to Open Source Projects with GeoLite Support

Jump to: Open Source Projects with GeoLite Support Go

Who is online

Users browsing this forum: No registered users and 0 guests

- Board index
- The team • Delete all board cookies • All times are UTC