

Subject: Sawmill Newsletter: Reducing Database Disk Usage

From: Jo Dee Koller <jodee@flowerfire.com>

Date: Thu, 15 Jul 2010 11:51:23 -0700

To: Greg Ferrar <ferrar@flowerfire.com>



Sawmill Newsletter

July 15, 2010

Welcome to the Sawmill Newsletter!

You're receiving this newsletter because during the downloading or purchase of Sawmill, you checked the box to join our mailing list. If you wish to be removed from this list, please send an email, with the subject line of "UNSUBSCRIBE" to newsletter@sawmill.net (please include the entire message, as the identifying information is at the bottom).

News

Sawmill 8.1.5 shipped on June 2, 2010. This is a bug-fix release--it fixes a number of bugs. This release is free to existing Sawmill 8 users. It is recommended for anyone who is experiencing problems with Sawmill 8.1.4 or earlier. You can download Sawmill 8.1.3 from <http://sawmill.net/download.html>.

Sawmill 7 users can upgrade to Sawmill 8 for half of the license price; or if you have Premium Support, the upgrade is free. Major features of Sawmill 8 include support for Oracle and Microsoft SQL Server databases, real-time reporting, a completely redesigned web interface, better multi-processor and multi-core support, and role-based authentication control.

This issue of the Sawmill Newsletter describes methods for reducing the disk usage of the database for a profile.

Get The Most Out Of Sawmill With Professional Services

Looking to get more out of your statistics from Sawmill? Running short on time, but need the information now to make critical business decisions? Our Professional Service Experts are available for just this situation and many others. We will assist in the initial installation of Sawmill using best practices; work with you to integrate and configure Sawmill to generate reports in the shortest possible time. We will tailor Sawmill to your environment, create a customized solution, be sensitive to your requirements and stay focused on what your business needs are. We will show you areas of Sawmill you may not even be aware of, demonstrating these methods will provide you with many streamlined methods to get you the information more quickly. Often you'll find that Sawmill's deep analysis can even provide you with information you've been after but never knew how to reach, or possibly never realized was readily available in reports. Sawmill is an extremely powerful tool for

your business, and most users only exercise a fraction of this power. That's where our experts really can make the difference. Our Sawmill experts have many years of experience with Sawmill and with a large cross section of devices and business sectors. Our promise is to very quickly come up with a cost effective solution that fits your business, and greatly expand your ROI with only a few hours of fee based Sawmill Professional Services. For more information, a quote, or to speak directly with a Professional services expert contact consulting@flowerfire.com.

Tips & Techniques: Reducing Database Disk Usage

Sawmill reads log data, stores it in a database, and generates reports from that database. This "back end database" contains virtually all the information from the original log files, and also contains a large amount of additional "derived" information: derived fields (like Country, derived from the IP address in the log data), cross-reference groups (for faster top-level report generation), and indices (for faster filtered report generation). This requires disk space, and if the log dataset is very large, then the amount of disk space required by the database can be very large. To some degree this is unavoidable: in order to report ad hoc on the original data with any combination of filters (which is what Sawmill does), all the information must be in the database, so the database will be as large as the input data, plus the derived information. In practice, the database may be two or four times larger than the uncompressed log data, when using default settings and full detail.

If disk space usage isn't an issue, then the database can be left in its default state, with default settings. But if disk space is tight, there are many ways to reduce the size of the database. The possibilities can be divided into two categories: trade performance for disk space, or trade information for disk space.

Category I: Trade Performance For Disk Space

This category of optimization generally eliminates some of the space-intensive caches and performance optimization structures of the database. Eliminating or simplifying these structures allows the database to use less disk space, but slows report generation.

I-1: Eliminate Indices. Indices are structures tied to database fields, which improve the performance of searching the database for specific values. They are used when filtering: for instance, if you zoom in on File Type=HTML, and then display a Top IPs report, the index on the File Type field will help Sawmill find and compute the report much faster. Indices can be turned off for any database field, in Config -> Database Fields, by unchecking the Index checkbox for that particular field. Unchecking an index will reduce the database size (after rebuild), but will slow reports filtered on that field.

I-2: Eliminate Cross-reference Groups (xrefs). Cross-reference groups are structures which precompute top-level reports. For instance, the File Types report typically has an associated cross-reference group, which makes that report very fast to load (in essence, the report values are computed when the database is built, rather than having to be queried from all rows of the database when the report is generated). Cross-reference Groups can be disabled or deleted in Config -> Cross Reference Groups. Deleting or disabling a cross-reference group will reduce the database size (after rebuild), but will slow top-level reports which use the fields in that group.

I-3: Flatten Cross-reference Groups. As with the above (Eliminate Cross-reference Groups), this involves cross-reference groups; but instead of completely eliminating them, it is makes them non-hierarchical. This can also be done in Config -> Cross Reference Groups, on a

group-by-group basis. Non-hierarchical xrefs take less disk space than hierarchical xrefs, but reports generated from them are slower, especially when those reports are displaying items which are not at the bottom of their hierarchy, like directories (in a "pages" report) or months, or the root (the Overview). So flattening some xref groups will reduce database size (on rebuild), but will slow some top-level reports.

I-4: Compress The File System. Some file systems, like NTFS (often used on Windows) have a compression option built into the file system. Compressing the Sawmill database folder, at the file system level, will shrink database usage by 50% or so, at no cost in completeness of the data. However, the performance effect of this can be very severe--the database may be several times slower if it's on a compressed system. Therefore, this option should only be used if performance is of very little importance.

Category II: Trade Information For Disk Space

This category of optimization eliminates information from the database. Since the database size is proportional to the amount of information in it, removing information will reduce the database size. However, depending on the information-reduction method used, some reports may not be available, or some details may be missing, which would be present in a fully complete database.

II-1: Tune The Size Of Database Fields. Most fields in Sawmill's database are represented as integers. Integers are represented as a sequence of bits: 8, 16, 32, or 64 bits. The more bits a field uses, the more disk space it uses in the database, and the higher the numbers it can represent. This is set for each database field in Config -> Database Fields. Sawmill's default behavior is to use as many bits for a field as the CPU supports directly--it uses 32 bits for all fields on a 32-bit system, or 64 bits on a 64-bit system. This is sometimes overkill, especially on 64-bit systems--64 bits can represent enormous numbers, but in a field like File Types, which usually has less than 100 values, 8 bits is sufficient. Reducing the File Types field to use 8 bits, reduces the disk usage of the File Type field from 64 bits to 8 bits (an 8x reduction), at no cost, as long as there are no more than 255 unique file types in the log data (if there are more, this will cause reporting errors, so be sure to keep the bits high enough to handle the requirement!). 8 bits can represent up to about 255; 16 bits up to about 65,000; 32 bits up to about 4 billion, and 64 bits can represent any number you're likely to encounter. So if you can put a limit on the number of unique values used for each database field (you can look at the number of rows in the corresponding report to get an estimate), then you can reduce the bits for that field to a smaller number, saving space. The same can work for numerical fields like "events" or "hits" or "page views," but in this case the field must be large enough to hold the *sum* of all field values--look in the Overview to see the actual sum, and if it's less than 4 billion, for instance, you can use 32 bits (just to be safe, use 64 bits if it's more than 1 billion).

II-2: Reduce Input Log Data. If the input data is a billion rows, and you remove half the files from the log source and rebuild, the database will be about half as large. So if some of the files in your log source contain information you don't care about--for instance if they're from a time period before what you care about, or if they're from a server you don't need to track--take those files out of the log source to reduce database size.

II-3: Filter Out Input Log Data. This method removes *specific* events from the database, by rejecting them during processing. For instance, a web server log with a billion lines might have 700,000 of those lines for non-page files like images, scripts, or style sheets. If all you care about are page views, you can add a log filter to reject all those events--the resulting database will have 300,000 rows, and will be about 70% smaller than the full database. In general, whenever there's a category of events you don't need to report on, you can filter it out to save space.

II-4: Remove Database Fields. A database contains many fields--for a proxy it might have a source IP, date/time, URL, and MIME type field, in addition to dozens of others. If there's a field you don't need to know about, remove it from the database, by deleting it in Config -> Database Fields (see the earlier newsletter, Deleting Database Fields, for information about deleting database fields and all their dependencies)--after

rebuilding, the database will have no information about that field, which will reduce the size. It is often possible to remove *many* fields, especially for formats which have dozens of fields by default, reducing the database to half its original size or less. To determine which fields to remove, start with the Single-Page Summary--look for report elements there you don't care about, or columns you don't care about, and remove the corresponding fields.

II-5: Remove/Expire Database Data. This is similar to reducing input log data, but after the fact. If you import a billion lines of data, and then use the "Remove Database Data" option in Admin -> Scheduler, to remove data from the database (typically, data older than a particular expiration date), the database will shrink. This is useful for keeping the database size in check by using a rolling window of data: a nightly database update to pull in the latest log data, plus a nightly database expiration to remove the oldest day of log data, maintains a database which has a roughly fixed size. It is critical to note, however, that database data removal temporarily creates a full copy of the main table of the database, and therefore uses up to the full size of the database in disk space--i.e., it temporarily *doubles* the database disk usage, before shrinking it. So when using a database update/remove cycle, you must not allow the available disk space to drop to less than the size of the largest database.

Professional Services

This newsletter describes methods for reducing the disk space used by a database. If you need assistance with this tuning, or with any other Sawmill tasks, our Sawmill Experts can help. Contact sales@sawmill.net for more information.

[Article revision v1.0]

[ClientID: 43726]