Molecular Systems Biology 7; Article number 539; doi:10.1038/msb.2011.75
Citation: ***Molecular Systems Biology* 7: 539**

www.molecularsystemsbiology.com

molecular systems biology

REPORT

Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega

Fabian Sievers[1,8], Andreas Wilm[2,8], David Dineen[1], Toby J Gibson[3], Kevin Karplus[4], Weizhong Li[5], Rodrigo Lopez[5], Hamish McWilliam[5], Michael Remmert[6], Johannes Söding[6], Julie D Thompson[7] and Desmond G Higgins[1,*]

[1] School of Medicine and Medical Science, UCD Conway Institute of Biomolecular and Biomedical Research, University College Dublin, Dublin, Ireland, [2] Computational and Systems Biology, Genome Institute of Singapore, Singapore, [3] Structural and Computational Biology Unit, European Molecular Biology Laboratory, Heidelberg, Germany, [4] Department of Biomolecular Engineering, University of California, Santa Cruz, CA, USA, [5] EMBL Outstation—European Bioinformatics Institute, Wellcome Trust Genome Campus, Hinxton, Cambridge, UK, [6] Gene Center Munich, University of Munich (LMU), Muenchen, Germany and [7] Département de Biologie Structurale et Génomique, IGBMC (Institut de Génétique et de Biologie Moléculaire et Cellulaire), CNRS/INSERM/Université de Strasbourg, Illkirch, France
[8] These authors contributed equally to this work
* Corresponding author. UCD Conway Institute of Biomolecular and Biomedical Research, University College Dublin, Belfield, Dublin 4, Ireland. Tel.: +353 1 716 6833; Fax: +353 1 716 6713; E-mail: des.higgins@ucd.ie

Received 23.7.11; accepted 6.9.11

Multiple sequence alignments are fundamental to many sequence analysis methods. Most alignments are computed using the progressive alignment heuristic. These methods are starting to become a bottleneck in some analysis pipelines when faced with data sets of the size of many thousands of sequences. Some methods allow computation of larger data sets while sacrificing quality, and others produce high-quality alignments, but scale badly with the number of sequences. In this paper, we describe a new program called Clustal Omega, which can align virtually any number of protein sequences quickly and that delivers accurate alignments. The accuracy of the package on smaller test cases is similar to that of the high-quality aligners. On larger data sets, Clustal Omega outperforms other packages in terms of execution time and quality. Clustal Omega also has powerful features for adding sequences to and exploiting information in existing alignments, making use of the vast amount of precomputed information in public databases like Pfam.

Molecular Systems Biology **7**: 539; published online 11 October 2011; doi:10.1038/msb.2011.75
Subject Categories: bioinformatics
Keywords: bioinformatics; hidden Markov models; multiple sequence alignment

Introduction

Multiple sequence alignments (MSAs) are essential in most bioinformatics analyses that involve comparing homologous sequences. The exact way of computing an optimal alignment between N sequences has a computational complexity of $O(L^N)$ for N sequences of length L making it prohibitive for even small numbers of sequences. Most automatic methods are based on the 'progressive alignment' heuristic (Hogeweg and Hesper, 1984), which aligns sequences in larger and larger subalignments, following the branching order in a 'guide tree.' With a complexity of roughly $O(N^2)$, this approach can routinely make alignments of a few thousand sequences of moderate length, but it is tough to make alignments much bigger than this. The progressive approach is a 'greedy algorithm' where mistakes made at the initial alignment stages cannot be corrected later. To counteract this effect, the consistency principle was developed (Notredame *et al*, 2000). This has allowed the production of a new generation of more accurate aligners (e.g. T-Coffee (Notredame *et al*, 2000)) but at the expense of ease of computation. These methods give 5–10% more accurate alignments, as measured on benchmarks, but are confined to a few hundred sequences.

In this report, we introduce a new program called Clustal Omega, which is accurate but also allows alignments of almost any size to be produced. We have used it to generate alignments of over 190 000 sequences on a single processor in a few hours. In benchmark tests, it is distinctly more accurate than most widely used, fast methods and comparable in accuracy to some of the intensive slow methods. It also has powerful features for allowing users to reuse their alignments so as to avoid recomputing an entire alignment, every time new sequences become available.

The key to making the progressive alignment approach scale is the method used to make the guide tree. Normally, this involves aligning all N sequences to each other giving time and memory requirements of $O(N^2)$. Protein families with $>50\,000$ sequences are appearing and will become common from various wide scale genome sequencing projects. Currently, the only method that can routinely make alignments of more than

about 10 000 sequences is MAFFT/PartTree (Katoh and Toh, 2007). It is very fast but leads to a loss in accuracy, which has to be compensated for by iteration and other heuristics. With Clustal Omega, we use a modified version of mBed (Blackshields *et al*, 2010), which has complexity of O(N log N), and which produces guide trees that are just as accurate as those from conventional methods. mBed works by 'emBedding' each sequence in a space of n dimensions where n is proportional to log N. Each sequence is then replaced by an n element vector, where each element is simply the distance to one of n 'reference sequences.' These vectors can then be clustered extremely quickly by standard methods such as K-means or UPGMA. In Clustal Omega, the alignments are then computed using the very accurate HHalign package (Söding, 2005), which aligns two profile hidden Markov models (Eddy, 1998).

Clustal Omega has a number of features for adding sequences to existing alignments or for using existing alignments to help align new sequences. One innovation is to allow users to specify a profile HMM that is derived from an alignment of sequences that are homologous to the input set. The sequences are then aligned to these 'external profiles' to help align them to the rest of the input set. There are already widely available collections of HMMs from many sources such as Pfam (Finn *et al*, 2009) and these can now be used to help users to align their sequences.

Results

Alignment accuracy

The standard method for measuring the accuracy of multiple alignment algorithms is to use benchmark test sets of reference alignments, generated with reference to three-dimensional structures. Here, we present results from a range of packages tested on three benchmarks: BAliBASE (Thompson *et al*, 2005), Prefab (Edgar, 2004) and an extended version of HomFam (Blackshields *et al*, 2010). For these tests, we just report results using the default settings for all programs but with two exceptions, which were needed to allow MUSCLE (Edgar, 2004) and MAFFT to align the biggest test cases in HomFam. For test cases with >3000 sequences, we run MUSCLE with the –maxiter parameter set to 2, in order to finish the alignments in reasonable times. Second, we have run several different programs from the MAFFT package. MAFFT (Katoh *et al*, 2002) consists of a series of programs that can be run separately or called automatically from a script with the *–auto* flag set. This flag chooses to run a slow, consistency-based program (L-INS-i) when the number and lengths of sequences is small. When the numbers exceed inbuilt thresholds, a conventional progressive aligner is used (FFT-NS-2). The latter is also the program that is run by default if MAFFT is called with no flags set. For very large data sets, the *–parttree* flag must be set on the command line and a very fast guide tree calculation is then used.

The results for the BAliBASE benchmark tests are shown in Table I. BAliBASE is divided into six 'references.' Average scores are given for each reference, along with total run times and average total column (TC) scores, which give the proportion of the total alignment columns that is recovered. A score of 1.0 indicates perfect agreement with the benchmark. There are two rows for the MAFFT package: MAFFT (auto) and MAFFT default. In most (203 out of 218) BAliBASE test cases, the number of sequences is small and the script runs L-INS-i, which is the slow accurate program that uses the consistency heuristic (Notredame *et al*, 2000) that is also used by MSAprobs (Liu *et al*, 2010), Probalign, Probcons (Do *et al*, 2005) and T-Coffee. These programs are all restricted to small numbers of sequences but tend to give accurate alignments. This is clearly reflected in the times and average scores in Table I. The times range from 25 min up to 22 h for these packages and the accuracies range from 55 to 61% of columns correct. Clustal Omega only takes 9 min for the same runs but has an accuracy level that is similar to that of Probcons and T-Coffee.

The rest of the table is mainly taken by the programs that use progressive alignment. Some of these are very fast but this speed is matched by a considerable drop in accuracy compared with the consistency-based programs and Clustal Omega. The weakest program here, is Clustal W (Larkin *et al*, 2007) followed by PRANK (Löytynoja and Goldman, 2008). PRANK is not designed for aligning distantly related sequences but at giving good alignments for phylogenetic work with special

Table I BAliBASE results

Aligner	Av score (218 families)	BB11 (38 families)	BB12 (44 families)	BB2 (41 families)	BB3 (30 families)	BB4 (49 families)	BB5 (16 families)	Tot time (s)	Consistency
MSAprobs	0.607	0.441	0.865	0.464	0.607	0.622	0.608	12 382.00	Yes
Probalign	0.589	0.453	0.862	0.439	0.566	0.603	0.549	10 095.20	Yes
MAFFT (auto)	0.588	0.439	0.831	0.450	0.581	0.605	0.591	1475.40	Mostly (203/218)
Probcons	0.558	0.417	0.855	0.406	0.544	0.532	0.573	13 086.30	Yes
Clustal Ω	0.554	0.358	0.789	0.450	0.575	0.579	0.533	539.91	No
T-Coffee	0.551	0.410	0.848	0.402	0.491	0.545	0.587	81 041.50	Yes
Kalign	0.501	0.365	0.790	0.360	0.476	0.504	0.435	21.88	No
MUSCLE	0.475	0.318	0.804	0.350	0.409	0.450	0.460	789.57	No
MAFFT (default)	0.458	0.258	0.749	0.316	0.425	0.480	0.496	68.24	No
FSA	0.419	0.270	0.818	0.187	0.259	0.474	0.398	53 648.10	No
Dialign	0.415	0.265	0.696	0.292	0.312	0.441	0.425	3977.44	No
PRANK	0.376	0.223	0.680	0.257	0.321	0.360	0.356	128 355.00	No
ClustalW	0.374	0.227	0.712	0.220	0.272	0.396	0.308	766.47	No

The figures are total column scores produced using bali score on core columns only. The average score over all families is given in the second column. The results for BAliBASE subgroupings are in columns 3–8. The total run time for all 218 families is given in the second last column. The last column indicates whether the method is consistency based.

attention to gaps. These gap positions are not included in these tests as they tend not to be structurally conserved. Dialign (Morgenstern *et al*, 1998) does not use consistency or progressive alignment but is based on finding best local multiple alignments. FSA (Bradley *et al*, 2009) uses sampling of pairwise alignments and 'sequence annealing' and has been shown to deliver good nucleotide sequence alignments in the past.

The Prefab benchmark test results are shown in Table II. Here, the results are divided into five groups according to the percent identity of the sequences. The overall scores range from 53 to 73% of columns correct. The consistency-based programs MSAprobs, MAFFT L-INS-i, Probalign, Probcons and T-Coffee, are again the most accurate but with long run times. Clustal Omega is close to the consistency programs in accuracy but is much faster. There is then a gap to the faster progressive based programs of MUSCLE, MAFFT, Kalign (Lassmann and Sonnhammer, 2005) and Clustal W.

Results from testing large alignments with up to 50 000 sequences are given in Table III using HomFam. Here, each alignment is made up of a core of a Homstrad (Mizuguchi *et al*, 1998) structure-based alignment of at least five sequences. These sequences are then inserted into a test set of sequences from the corresponding, homologous, Pfam domain. This gives very large sets of sequences to be aligned but the testing is only carried out on the sequences with known structures. Only some programs are able to deliver alignments at all, with data sets of this size. We restricted the comparisons to Clustal Omega, MAFFT, MUSCLE and Kalign. MAFFT with default settings, has a limit of 20 000 sequences and we only use MAFFT with –parttree for the last section of Table III. MUSCLE becomes increasingly slow when you get over 3000 sequences. Therefore, for >3000 sequences we used MUSCLE with the faster but less accurate setting of –maxiters 2, which restricts the number of iterations to two.

Overall, Clustal Omega is easily the most accurate program in Table III. The run times show MAFFT default and Kalign to be exceptionally fast on the smaller test cases and MAFFT –parttree to be very fast on the biggest families. Clustal Omega does scale well, however, with increasing numbers of sequences. This scaling is described in more detail in the Supplementary Information. We do have two further test cases with $>50\,000$ sequences, but it was not possible to get results for these from MUSCLE or Kalign. These are described in the Supplementary Information as well.

Table III gives overall run times for the four programs evaluated with HomFam. Figure 1 resolves these run times case by case. Kalign is very fast for small families but does not scale as well. Overall, MAFFT is faster than the other programs over all test case sizes but Clustal Omega scales similarly. Points in Figure 1 represent different families with different average sequence lengths and pairwise identities. Therefore, the scalability trend is fuzzy, with larger dots occurring generally above smaller dots. Supplementary Figure S3 shows scalability data, where subsets of increasing size are sampled from one large family only. This reduces variability in pairwise identity and sequence length.

External profile alignment

Clustal Omega can read extra information from a profile HMM derived from preexisting alignments. For example, if a user

Table II Prefab results

Aligner	$0<$ %ID$\leqslant$100 (1682 families)	0 $\leqslant$ %ID $\leqslant$ 20 (912 families)	20 $\leqslant$ %ID $\leqslant$ 40 (563 families)	40 $\leqslant$ %ID $\leqslant$ 70 (117 families)	70 $\leqslant$ %ID $\leqslant$ 100 (90 families)	Total time (s) (1682 families)	Consistency
MSAprobs	0.737	0.591	0.889	0.965	0.971	51 286.00	Yes
MAFFT (auto)	0.721	0.569	0.876	0.961	0.979	4544.45	Yes
Probalign	0.719	0.563	0.881	0.961	0.977	35 117.30	Yes
Probcons	0.717	0.562	0.876	0.955	0.972	46 908.30	Yes
T-Coffee	0.710	0.558	0.865	0.950	0.972	175 789.00	Yes
Clustal Ω	0.700	0.535	0.866	0.967	0.980	1698.06	No
MUSCLE	0.677	0.507	0.850	0.946	0.976	2068.56	No
MAFFT	0.677	0.513	0.836	0.961	0.979	225.56	No
Kalign	0.649	0.474	0.817	0.957	0.979	80.81	No
ClustalW2	0.617	0.430	0.797	0.933	0.975	3433.53	No
Dialign	0.595	0.398	0.783	0.940	0.974	18 909.70	No
PRANK	0.586	0.390	0.767	0.951	0.978	351 498.00	No
FSA	0.534	0.277	0.791	0.965	0.976	229 391.00	No

Total column scores (TC) are shown for different percent identity ranges; the second column is the average score over all test cases. The total run time in seconds is shown in the second last column. The last column indicates if the method is consistency based.

wishes to align a set of globin sequences and has an existing globin alignment, this alignment can be converted to a profile HMM and used as well as the sequence input file. This HMM is here referred to as an 'external profile' and its use in this way as 'external profile alignment' (EPA). During EPA, each sequence in the input set is aligned to the external profile. Pseudocount information from the external profile is then transferred, position by position, to the input sequence. Ideally, this would be used with large curated alignments of particular proteins or domains of interest such as are used in metagenomics projects. Rather than taking the input sequences and aligning them from scratch, every time new sequences are found, the alignment should be carefully maintained and used as an external profile for EPA. Clustal Omega also can align sequences to existing alignments using conventional alignment methods. Users can add sequences to an alignment, one by one or align a set of aligned sequences to the alignment.

In this paper, we demonstrate the EPA approach with two examples. First, we take the 94 HomFam test cases from the previous section and use the corresponding Pfam HMM for EPA. Before EPA, the average accuracy for the test cases was 0.627 of correctly aligned Homstrad positions but after EPA it rises to 0.653. This is plotted, test case for test case in Figure 2A. Each dot is one test case with the TC score for Clustal Omega plotted against the score using EPA. The second example is illustrated in Figure 2B. Here, we take all the BAliBASE reference sets and align them as normal using Clustal Omega and obtain the benchmark result of 0.554 of columns correctly aligned, as already reported in Table I. For EPA, we use the benchmark reference alignments themselves as external profiles. The results now jump to 0.857 of columns correct. This is a jump of over 30% and while it is not a valid measure of Clustal Omega accuracy for comparison with other programs, it does illustrate the potential power of EPA to use information in external alignments.

Iteration

EPA can also be used in a simple iteration scheme. Once a MSA has been made from a set of input sequences, it can be converted into a HMM and used for EPA to help realign the input sequences. This can also be combined with a full recalculation of the guide tree. In Figure 3, we show the results of one and two iterations on every test case from HomFam. The graph is plotted as a running average TC score for all test cases with N or fewer test cases where N is plotted on the horizontal axis using a log scale. With some smaller test cases, iteration actually has a detrimental effect. Once you get near 1000 or more sequences, however, a clear trend emerges. The more sequences you have, the more beneficial the effect of iteration is. With bigger test cases, it becomes more and more beneficial to apply two iterations. This result confirms the usefulness of EPA as a general strategy. It also confirms the difficulty in aligning extremely large numbers of sequences but gives one partial solution. It also gives a very simple but effective iteration scheme, not just for guide tree iteration, as used in many packages, but for iteration of the alignment itself.

Discussion

The main breakthroughs since the mid 1980s in MSA methods have been progressive alignment and the use of consistency. Otherwise, most recent work has concerned refinements for speed or accuracy on benchmark test sets. The speed increases have been dramatic but, with just two major exceptions, the methods are still basically O(N^2) and incapable of being extended to data sets of >10 000 sequences. The two exceptions are mBed, used here, and MAFFT PartTree. PartTree is faster but at the expense of accuracy, at least as judged by the benchmarking here. The second group of recent developments

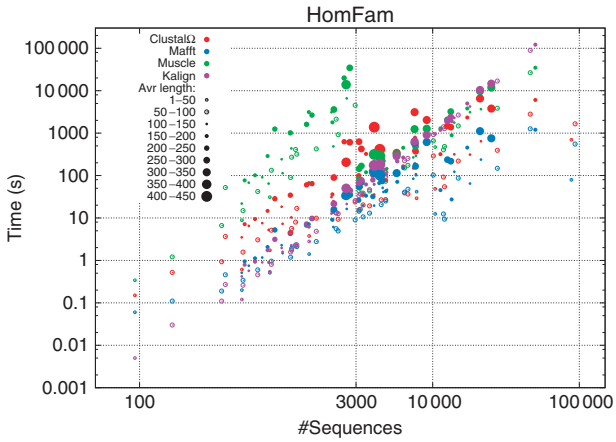

Figure 1 Alignment time for Clustal Omega (red), MAFFT (blue), MUSCLE (green) and Kalign (purple) against the number of sequences of HomFam test sets. Average sequence length is rendered by point size. Both axes have logarithmic scales. Clustal Omega and Kalign were run with default flags over the entire range. MUSCLE was run with –maxiters 2 for $N>3000$ sequences. MAFFT was run with –parttree for $N>10\,000$ sequences.

Table III HomFam benchmarking results

	$93 \leqslant N \leqslant 2957$ (41 families)	$3127 \leqslant N \leqslant 9105$ (33 families)	$10\,099 \leqslant N \leqslant 50\,157$ (18 families)
Aligner	TC/t (s)	TC/t (s)	TC/t (s)
Clustal Ω	0.708/2114.0	0.639/11 719.5	0.464/27 328.9
Kalign	0.569/324.9	0.563/6752.0	0.420/286 711.0
MAFFT default	0.550/238.9	0.462/3115.4	–/–
MAFFT –parttree	–/–	–/–	0.253/6119.4
MUSCLE default	0.533/104 587.0	–/–	–/–
MUSCLE –maxiters 2	–/–	0.416/8239.2	0.216/110 292.0

The columns show total column score (TC) and total run time in seconds for groupings of small (<3000 sequences), medium (3000–10 000 sequences) and large (>10 000 sequences) HomFam test cases.

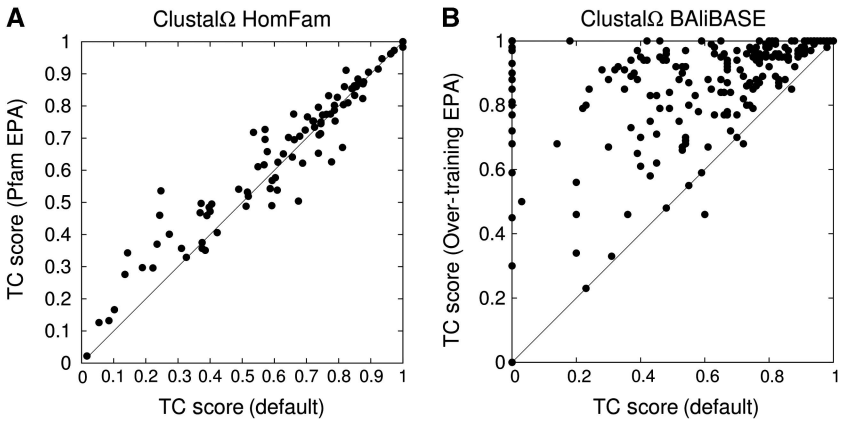

Figure 2 EPA for HomFam and BAliBASE. Points represent TC scores of Clustal Omega alignment with EPA versus TC scores of default Clustal Omega alignment (without EPA). Points above bisectrix represent beneficial effect of EPA, points below deleterious effect. Average improvement in (**A**) 2.5%. HMMs taken from Pfam, benchmarking carried out using corresponding structure-based alignment in Homstrad. Average improvement in (**B**) over 30%. Here, test sets and EPA-HMMs were both derived from BAliBASE reference alignments.

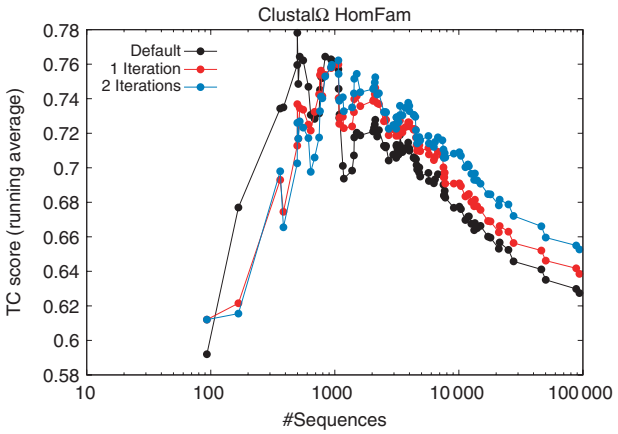

Figure 3 Iteration of HomFam alignments. Points represent cumulative running averages of TC scores. Clustal Omega default results in black, results after 1 iteration in red, after 2 iterations in blue. Iterations are combined HMM/guide tree iterations; *x* axis, logarithmic and *y* axis, linear scale.

have concerned accuracy. This has tended to focus on results from benchmarking, a potentially contentious issue (Aniba *et al*, 2010; Edgar, 2010). The benchmark test sets that we have are limited in scope and heavily biased toward single domain globular proteins. This has the potential to lead to methods that behave well on benchmarks but which are not so flexible or useful in real-world situations. One development to improve accuracy has been the recruitment of extra homologs to bulk up input data sets. This seems to work well with the consistency-based methods and for small data sets. It appears, however, that there is a limit to the extra accuracy that can be obtained this way, without further development. The extra sequences may also bring in noise and dramatically increase the complexity of the computational problem. This can be partly fixed by iteration but, EPA to a high-quality reference alignment might be a better solution. This also raises the need for methods to visualize such large alignments, in order to detect problems. A second major focus for development has been the use of external information such as RNA structure (Wilm *et al*, 2008) or protein structure predictions (Pirovano *et al*, 2008).

EPA is a new approach that allows users to exploit information in their own or in publicly available alignments. It does not force new sequences to follow the older alignment exactly. The new sequences get aligned to each other using progressive alignment but the information in the external profile can help provide information as to which amino acids are most likely to occur at each position in a sequence. Most methods attempt to predict this from general models of protein evolution with secondary structure prediction as a refinement. In this paper, we have shown that even using the mass produced alignments from Pfam as external profiles provides a small increase in accuracy for a large general set of test cases. This opens up a new set of possibilities for users to make use of the information contained in large, publicly available alignments and creates an incentive for database providers to make very high-quality alignments available.

One of the reasons for the great success of Clustal X was the very user-friendly graphical user interface (GUI). This, however, is not as critical as in the past due to the widespread availability of web-based services where the GUI is provided by the web-based front-end server. Further, there are several very high-quality alignment viewers and editors such as Jalview (Clamp *et al*, 2004) and Seaview (Gouy *et al*, 2010) that read Clustal Omega output or which can call Clustal Omega directly.

Materials and methods

Clustal Omega is licensed under the GNU Lesser General Public License. Source code as well as precompiled binaries for Linux, FreeBSD, Windows and Mac (Intel and PowerPC) are available at http://www.clustal.org. Clustal Omega is available as a command line program only, which uses GNU-style command line options, and also accepts ClustalW-style command options for backwards compatibility and easy integration into existing pipelines.

Clustal Omega is written in C and C++ and makes use of a number of excellent free software packages. We used a modified version of

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.