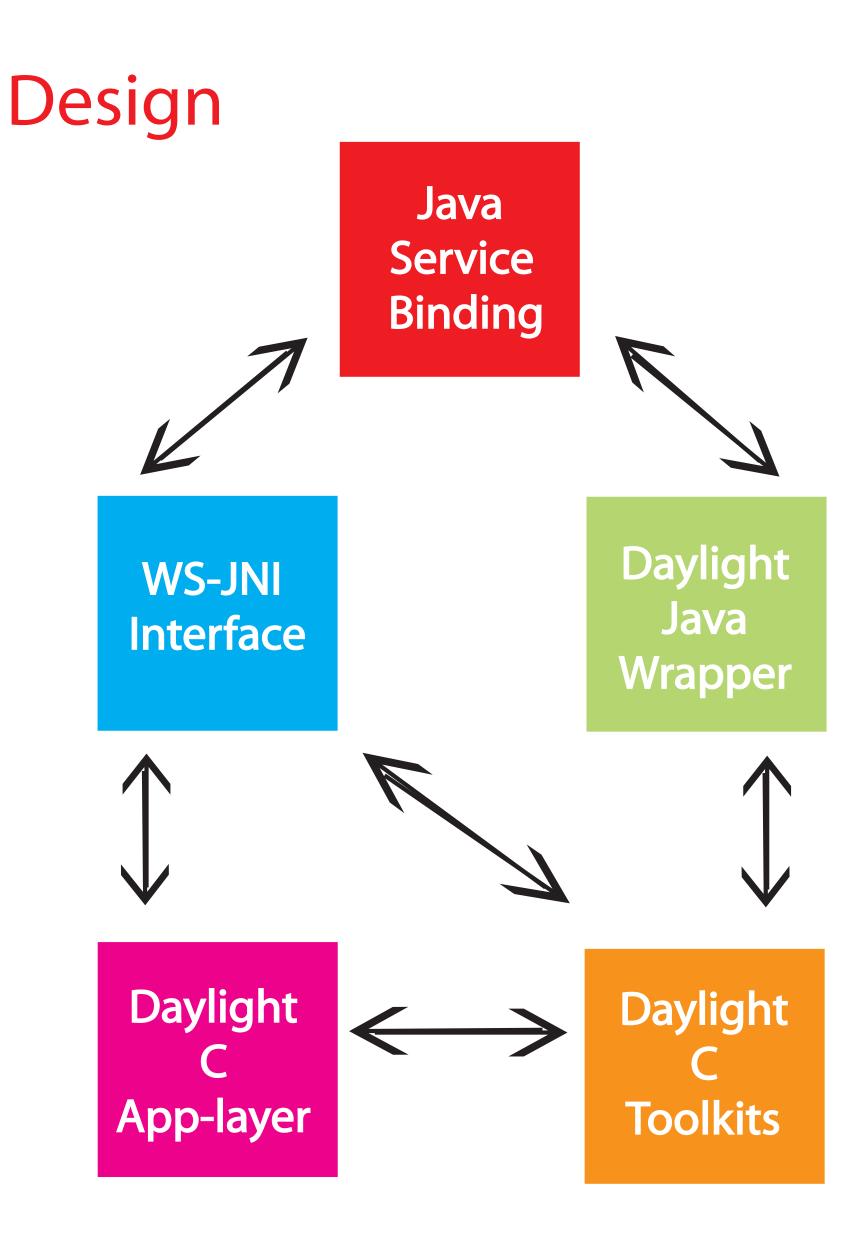
Daylight Web Services

Network Access to **Chemistry-Oriented Processes**

Daylight Web Services provide a valuable mechanism for delivering cheminformatics functionality within an organization. Web services use standard, open protocols to pass parameter data along with a particular service request over a network, thus triggering an action and the return of a defined response. Daylight Web Services use standard protocols with SOAP messaging to provide easy access to Daylight functions and are ready to use in user-designed workflows, web-based applications, or command line applications.

SOAP-Based Services

- Easy to deploy over a site or global enterprise
- Open, object-oriented interface
- Use WSDL (XML) to provide programmer API
- Provides high-level, prepackaged functionality



Prepared by

Terence Brunck, Ph. D, VP - Research & Development, Daylight CIS, Inc. - terry@daylight.com Scott Dixon, Ph. D, Senior Scientist, Daylight CIS, Inc. - scott@daylight.com Peter Nielsen, Director - Marketing & Business Development, Daylight CIS, Inc. - peter@daylight.com

Current Daylight Services

canonicalizeSmiles - parses a list of molecules or reactions and generates the corresponding canonical SMILES

getDepiction - converts a single SMILES string or TDT into a structural diagram in GIF format

getProperties - calculates values for a specified list of different physical properties for one or more input SMILES

CONVERTSTRUCTURE - interconverts data and structures between MDL chemical table-based file formats and Daylight SMILES-based formats

deriveSDClusters - partitions moderately-sized sets of input SMILES into clusters with significant scaffolds

deriveScaffold - generates a single SMARTS[®] scaffold that captures all common substructure elements including ring topology from a list of SMILES

Web Services in Development

getClogP - Biobyte logP calculation

transformReactants - generates reactions and products

generateTautomers - compute "canonical" or all tautomers

normalizeStructure - convert structure to "standard" form using business rules (desalt, etc.)

generateRTable - make an R-table from a scaffold and a set of molecules

checkLicenses - check the availability of Daylight licensed web services

Common Substructure Find scaffold for a data set

The common substructure function, now available both as a Daylight Web Service and as part of the Daylight Toolkit Package, allows one to take a set of structures represented as SMILES[™] and generates a SMARTS[®] scaffold that captures all common substructure elements of that set.

In constrast to a traditional maximum common substructure (MCS), the SMARTS[®] scaffold may contain multiple fragments of common substructure. The user can control the level of topology used (with or without ring bond counts) and a minimum fragment size to eliminate fragments of the scaffolds that are smaller than the minimum.

Scaffold-Directed Clustering Clusters characterized by a common substructure

The new scaffold-directed clustering is available both as a Daylight Web Service and as part of the Daylight Clustering Package. It is distinct from many other clustering methods in that it generates clusters of compounds, where each cluster is represented by a scaffold. The scaffold is a SMARTS[®] expression with one or more fragments that contains the structural features common to all members of the cluster. Clustering is directed in such a way as to guarantee a defined level of scaffold coverage for all members of a cluster.

It is an ideal clustering method for applications involving smaller data sets such as determining structure-activity relationships within lead optimization projects or analysis of structures provided in publications or patents.

The method differs from tree-based scaffold analysis methods in that 1) it does not use heuristics to generate scaffolds, 2) it does not use any predefined fragments, and 3) it generates a single set of clusters that meet an intuitive coverage metric. The scaffold-directed method is therefore more general and directly provides sets of compounds ready for further analysis.

DAYLGHT Chemical Information Systems, Inc.

Implementation

The ease of Java integration with the well-tested performance of C. All of the current web services are developed in Java and deployed under Apache Tomcat/Axis as war files and Daylight C-libraries. The service bindings rely on JNI calls to access Daylight's core code base. The C-level code base is thread-safe. All services return appropriate warning and error messages.

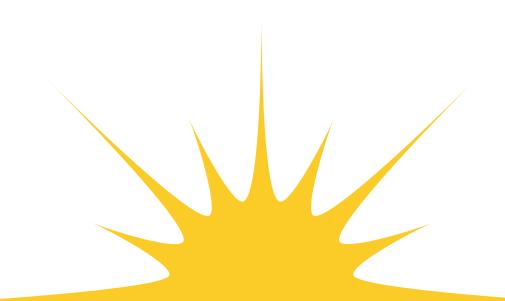
Server Set-up

- Install Daylight software distribution.
- Install Apache Tomcat.

- Make sure privileges are set correctly.

Clients

- Copy Daylight WSDL to your IDE.
- Write your app from the stubs, OR
- Pipeline Pilot, Inforsense).



• Copy Daylight war file to tomcat/webapps. • Set Tomcat environment variables to point to Daylight distribution.

• Set up your favorite IDE (eclipse, Netbeans, ...). • Build stubs for desired client type (JSP, commandline, ...).

Integrate into your third-party workflow applications (ie. Spotfire,

