Executive Summary

As a partner in our business, in our life, we want the computer to understand what we want, to understand what we mean, when we interact with it. Exalead CloudView uses the best web technologies to create enterprise information systems that exploit all your business data resulting in the most reactive, intelligent systems on the market.

Semantic processing covers a wide range of technologies that bridge the gap between text and meaning, between images and meaning. Semantic processing is the key to providing value to your business applications. Connecting information in a meaningful way is necessary to solve today’s management challenges:

- Collect all information available about a subject whatever it may be. This extends the reach of business applications to all data including unstructured data.
- Comprehend the sentiments of your customers regarding a product.
- Build proactive systems that react quickly to all new data.
- Give the right answer in the right time to your employees, by understanding the query they asked, in context.

Semantics is the study of meaning. In this white paper, we show how computers can deal with the semantics of texts, images and videos.

- We introduce semantic processing - *What can Semantics do for you?* on page 5
- We introduce Exalead CloudView’s semantic functionality and architecture - *CloudView Semantic Platform* on page 7
- We detail our natural language processing - *Natural Language Processing* on page 11
- We detail our semantic processing for classification - *Semantics for Content and Classification* on page 13
- We detail our semantic processing for search presentation - *Semantics for Search Presentation* on page 20
- We summarize the semantic features supported by Exalead - *CloudView Semantic Features* on page 27
We Welcome your Feedback

Whatever your role - Search platform user, IT analyst, business decision maker, system architect, security expert, or simply a curious reader - your feedback is important to us. We invite you to contact us at the addresses below with your comments, suggestions or questions.

Pour contacter Exalead
Gregory Grefenstette, Scientific Director
Tel: +33 (0) 1 55 35 26 26
ggrefens@exalead.com
www.exalead.com
Table of Contents

1. What can Semantics do for you? .................................................. 5
   1.1. Semantics ........................................................................... 5
   1.2. Semantic Search ................................................................. 6

2. CloudView Semantic Platform .................................................... 7
   2.1. What semantic functionality do we support? ......................... 7
   2.2. How do we implement semantics? ....................................... 8
   2.3. What is the Mining of Text (MOT) Architecture? ................ 8

3. Natural Language Processing .................................................... 11
   3.1. Language Identification ..................................................... 11
   3.2. Tokenization, Segmentation ................................................. 11
   3.3. Part of Speech Tagging and Grammatical Parsing ................. 12

4. Semantics for Content and Classification .................................. 13
   4.1. Named Entity Extraction .................................................... 13
   4.2. Contextual Entity Matching ................................................. 15
   4.3. Categorization ................................................................ 15
   4.4. Clustering ...................................................................... 16
   4.5. Sentiment Analysis ............................................................ 16
   4.6. Event Extraction and Triples ............................................. 17
   4.7. Business Item Extraction ................................................... 18

5. Semantics for Search Presentation ............................................. 20
   5.1. Linguistic Operators ......................................................... 20
   5.2. Stemming ...................................................................... 21
   5.3. Spell Checking ................................................................. 22
   5.4. Related Terms ................................................................ 23
   5.5. Summarization ................................................................. 23
   5.6. Question Answering ......................................................... 24
   5.7. Faceted Analysis .............................................................. 25

6. CloudView Semantic Features .................................................. 26
   6.1. Natural Language Processing ............................................. 26
   6.2. Semantic processing ......................................................... 26
1 What can Semantics do for you?

1.1 Semantics

Semantics processing of text means:

- Being able to tell when two different pieces of text are talking about the same thing.
- Being able to say the type, or apply some label, to something found in a text.

These two functions of matching and labeling can be done for words, phrases, sentences, paragraphs, documents and collections of documents. The area of computing dealing with semantic analysis of text is called Natural Language Processing. Natural language is language used by humans to communicate to humans, as opposed to artificial languages like computer programs, used to control a machine. Natural language is ambiguous since humans can interpret things in context.

Programming languages are exact. When a computer tries to understand the semantics of text, for example match two things or label something, it can make mistakes because a computer program does not have very much context.

Computer scientists have been trying to make computers do better and better jobs. These scientists have split the problem of semantic analysis into sub-tasks, and have tried to devise programs to perform each task, maybe not perfectly, but better and better. Some of these tasks are:

- Language identification: labeling the language in which a text is written (English, French, Dutch, ...).
- Genre identification: labeling who the text was written for (children, ordinary people, scientists), or the type of a text (e-mail, blog entry, newspaper article, annual report, etc.).
- Tokenization: splitting a text into individual words and sentences. For example, is the expression “i.e.” one token or two, or four? These are the things a program has to decide, but humans may disagree on this.
- Morphological analysis, stemming: recognizing when two different word forms come from the same root word. Words like “dog” and “dogs” are pretty easy, but what about think and thought?
- Grammatical analysis, part-of-speech tagging: labeling when a word is used as a noun, verb, adjective, adverb, etc. This is important because you might want to consider the word “thought” like “think” when it is a verb, but like “thoughts” when it is a noun.
- Word sense disambiguation: is “crane” in a given text a type of bird or a type of machine.
- Named entity extraction: what people, organizations, places are found in a text?
- Parsing, dependency analysis: identifying noun phrases and verb phrases, identify relations between words in a sentence such as the subjects or objects of verbs.
- Summarization: does this shorter text contain most of the same information as this longer text?
- Pronoun resolution: Who does this word “they” refer to in a text?
- Event and fact extraction: Is an attack mentioned in this text, even if the word “attack” was not used? If so, when did it take place, who was involved, was anybody hurt?
- Sentiment Analysis: Is this text positive or negative about something?
- Categorization: Is this a medical text, or financial, or sports?

These and other natural language processing tasks all involve deciding when two things are the same, or applying some label to a piece of text. All these tasks can be placed under the label of semantic analysis.

For multimedia data, semantic processing means not only extracting meaning from any text associated with images, sound or video, but also extracting information from the multimedia data: words from speech, and object recognition, such as face detection, from images.
1.2 Semantic Search

Natural Language Processing in a search engine allows the engine to decide some of the semantics of things found in ordinary text. The elements that turn search into Semantic Search are the abilities to match the same items in different forms. Text indexed by semantic search engines is enriched while they are being indexed, and queries receive the same treatment as soon as the user types them.

When we use databases, the semantics of a data item is defined by the field that the data is stored in. Databases often have fields like Lastname, FirstName, Salary in which the meaning of each entry is unambiguous.

Modern search engines like Exalead CloudView provide database connectors that allow a user to search databases without learning database query syntax. Searching in databases provides part of semantic search of a system like Exalead CloudView.

Modern search engines, like Exalead CloudView, provide hybrid searching over typed information extracted from structured databases, as well as searching over unstructured text.

When searching for typed objects in unstructured text, things become messier. Without clean-cut database field labels, it is no longer evident to the computer exactly what things mean. Incorporating Natural Language Processing into a search engine allows the engine to decide some of the semantics of things found in ordinary text. The elements that turn search into Semantic Search are the abilities to match the same items in different forms. Text indexed by semantic search engines is enriched while they are being indexed, and queries receive the same treatment as soon as the user types them. The enrichment allows the search engine to place the query in context, and to match query on document on more than just strings of characters. For example, a semantic search engine will know that "Rafael Nadal" is associated with "Tennis".

Stand-alone search is only one of the uses of search. All forms of information access become more intelligent when semantic processing is available for understanding text. Components such as Exalead Cloudview allow applications that include search on unstructured text to interact with structured data as well.
2 CloudView Semantic Platform

2.1 What semantic functionality do we support?

At the foundation of Exalead’s products and solutions is a high-performance unified information access platform: Exalead Cloudview.

With Exalead Cloudview, business users can simultaneously access information on their PCs, network servers and the Web without any additional tuning or integration. System administrators benefit from an architecture designed to leverage existing IT architectures. The system includes built-in information source connectors for the most common file formats to facilitate quick and easy integration with current information management systems. The system’s performance is aided by the use of innovative statistical linguistic techniques, which enable Exalead Cloudview to return more relevant results. Enterprise users can also increase the accuracy of their search results with advanced technologies including full-text search, unified search and multi-language search (European, Arabic, Chinese, …). Additionally, the full integration of natural language processing technologies enables unique indexing and query processing techniques that give the system important semantic processing capabilities.

Exalead Cloudview also provides the core infrastructure of both Exalead’s enterprise search products and its public search engine at www.exalead.com, the world class Web engine with a current index of 8 billion documents, the largest in Europe.

Beyond simple indexing, Exalead Cloudview can analyze and enrich data. The system can use statistical analysis to classify or annotate any kind of document, so that even unstructured plain text documents can be exploited by the system’s patented, assisted navigation features. In addition, this patented process of classifying documents, detecting related terms and generating navigational menus for results is fully automatic and unmanaged (though still transparent and customizable). One result is the extraction of new meaning and relationships from data, and the ability to organize search results into intuitive knowledge dashboards. An example for search is illustrated above.
With rich content previews and intuitive results navigation, such dashboards give users a snapshot view of available resources, letting them drill down on details or broaden their exploration with a click of the mouse.

2.2 How do we implement semantics?

Natural language and semantic processing involve transforming text in order to recognize more than the words that appear in the text. The input text is enriched, acquiring more information at each step in the processing. Former approaches to semantic processing were often small-scale, dealing with limited domains, like weather reports, and large monolithic programs that were difficult to modify and extend. Modern approaches break down the processing into coherent, self contained modules that can be strung together like sections of pipe, the output of one section feeding into the next section as input.

At Exalead we have developed a pipeline approach for semantic annotation of text that allows us to specify the degree of annotation that we need for a given application. We call this our Mining-Of-Text Architecture. Each step in the pipeline involves transforming the output of the previous step, usually by adding new annotations onto the units that have been identified in the text. Many of these steps involve using language specific resources, such as dictionaries, rules, or lists.

2.3 What is the Mining of Text (MOT) Architecture?

The low-level natural language processing and the semantic language processing are performed using the Mining-of-Text (MOT) architecture. This MOT architecture uses natural language and semantic processors that are applied sequentially in a pipeline with access to a set of linguistic resources that adapt intelligently to your business corpus. The MOT architecture ensures that the linguistic resources are updated dynamically to become a rich source of business data for the optimization of your applications. The content is annotated to efficiently extract meaningful information from unstructured data. What do annotations do?

- Allow dynamic categorization which encourages exploration of related content.
- Extract named entities, embedded meanings, related terms and relationships to help results navigation.
- Perform exact or fuzzy ontology matching to optimize business applications.
- Extract linguistic data to improve the efficiency of search through approximation, spell-checking and phonetization.

2.3.1 Why use the MOT?

The MOT architecture shares some of the objectives of the GATE and UIMA open source frameworks. What are the aims?

- Extract very complex semantic information (concepts, relations between concepts,...) to optimize the addition of structure to unstructured data.
- Provide a framework to integrate external software thus reducing the time to market for new semantic functionality.

The MOT architecture was developed with the same philosophy as the Exalead CloudView platform. As it is integrated in the Exalead CloudView distributed architecture it is resilient and scalable. Both are optimized in terms of performance and manageability. In terms of manageability, MOT has reduced the number of annotations and meta-information layers to use, so that it can be usable by non NLP-experts. MOT has been built to scale to Terabytes sized
corpus. This is typically an order of magnitude greater than the open source framework. It is able to scale to billions of words, and handle streams of hundreds of thousands of tokens per seconds. MOT has been tested against several billions of web pages.

2.3.2 Which components use the MOT?

The MOT pipeline architecture is used both in the indexing and the query pipeline. The input of a native pipe is a chunk of text with some associated information, for example language. This chunk must be converted into an annotated token stream that will contain tokenizer results and all associated annotations. The native pipe architecture handles documents as an annotated token stream for performance purposes.

When indexing a text corpus, Exalead CloudView uses the MOT architecture to tokenize, lemmatize, extract semantic information such as named entities or related terms, perform ontology matching, construct linguistic resources, categorize and cluster documents. Here is a simple example of a native pipe for indexing:
When parsing a query, **Exalead CloudView** uses the MOT architecture to tokenize, lemmatize, expand the query using semantic processors for phonetization, approximation, spell-checking, use linguistic resources to create navigation axes for categories based on named entities or ontologies or related terms. Here is a simple example of a native pipe for search:

![Pipeline for Search](image)

### 2.3.3 How do you configure the MOT?

When you deploy **Exalead CloudView**, the MOT architecture is totally configurable. This means that the semantic processing and the generation of the linguistic resources is 100% flexible.

- The tokenization and normalization configuration is common to both the indexing and query processing pipelines.
- You can configure the natural language processors and semantic processors that are used during indexing. For each processors you can configure the resources and the rules.
- You can configure the linguistic resources that are generated.
- You can configure the natural language processors and semantic processors that are used during query processing.
3 Natural Language Processing

Exalead CloudView is packaged with an extensive list of natural language processors which you can configure in the MOT. Each processor can enrich the document using all the information extracted by previous processors.

The natural language processors available are:

- Language detection.
- Tokenization and normalization, sentence boundary recognition for indexing and search.
- Stemming for indexing and search.
- Lemmatization, morphological and syntactic processors for indexing and search.
- Part of speech tagging.
- In addition to the processors listed above there are processors that are automatically deployed during indexing when you activate the generation of certain linguistic resources.

3.1 Language Identification

Most processing steps in extracting the semantics of a text are specific to the language the text is written. Exalead CloudView currently recognizes the principal 54 different languages used on the Internet. Some languages are written in varying codesets, agreements of assigning characters to different bytes, and Exalead CloudView also recognizes these different coding standards.

Language identification is easily performed once you have a few lines of text.

Using known language texts, we gather statistics of common character sequences, and use them to guess the language of a new text.

For example, the sequence –ck is more common to English than to French, and –que is more common to French texts than to English. Using statistics about these events, the Exalead language model affects a probability for each language given a new input text. The most probable language is then selected, or none, if the language detector cannot guarantee with an acceptable level of certainty that the language detected was the right one.

The language of a document is one aspect of Exalead CloudView’s treatment that is visible on our search engine website, www.exalead.com.

3.2 Tokenization, Segmentation

To identify the individual words contained in a text segment, the tokenizer must first understand the way that language separates words, for instance, the way English uses punctuation and white space to separate words. This is no trivial task. For example, in some cases, a ‘.’ may occur as a period marking the end of a sentence, or as part of a word, for example, in http://www.exalead.com. But the real challenge comes when handling languages such as Chinese or Japanese: spaces are not used to split sentences into words, so the tokenizer must use elaborate and expensive techniques and comprehensive dictionaries to be able to split a sentence while limiting the number of errors. In addition to parsing the indexed text or user queries into words, the Exalead tokenization module associates additional information with indexed text, thus adding linguistic metadata that can improve the relevancy or search results.

Certain European languages pose segmentation problems. Once a token has been recognized, we may have to further break it down into meaningful subparts. For example, in order to get to the meaning of the German word “produktivitätsverbesserungen” (productivity improvements) the computer needs to recognize the two parts of the agglutinated word. The problems of segmentation of words into meaningful subparts are also found in Finnish, Hungarian, and Spanish.
The internal tokenization capabilities of the Exalead CloudView platform address the needs of most organizations, properly balancing processing complexity and system performance. The open API gives the Exalead CloudView administrator the option to provide specialized language converters which can tokenize text according to any special processing needs for their application.

3.3 Part of Speech Tagging and Grammatical Parsing

Words play different roles in different contexts. In English as in most other languages, some words can be a noun or verb, depending on how they are used. Common parts of speech are things like noun, verb, adjective, adverb, article, pronoun. Having a computer decide what role a word is playing in a given sentence is called part-of-speech tagging. This tagging adds additional labels to the input tokens, the part of speech. During a preceding step of morphological analysis, all the roles that a word can play are added to a token, usually using a dictionary to look up each word’s possible parts-of-speech. Then a large statistical model is applied to all the words in a sentence to rank the most likely sequence of part-of-speech for the given words.\(^1\) Exalead CloudView integrates the language model of possible tag and word sequences for the major world languages. Once a tag has been chosen in context, a more accurate word normalization procedure, called lemmatization, associates the dictionary entry form of the word (rather than just a stem) to each token in the text.

Grammatical parsing will apply patterns of part of speech tags to delineate verb phrases and noun phrases. Other patterns of parsing will connect nearby phrases and decide on subjects of verbs and objects of verbs. These steps of parts of speech tagging and parsing provide annotations that can be exploited by the higher level functions we discuss in the next section.

---

4 Semantics for Content and Classification

Exalead CloudView is packaged with an extensive list of semantic processors that you can configure in the MOT. Each processor can enrich the document using all the information extracted by previous processors. The semantic processors available for indexing are:

- Named entities extraction.
- Ontology matching.
- Identity matching.
- Categorization.
- Clustering.
- Query matching.
- Related terms detection and extraction.
- In addition to the processors listed above there are semantic processors that are automatically deployed during indexing when you activate the generation of certain linguistic resources.

4.1 Named Entity Extraction

Named entities are examples of typed information that can be found by applying linguistic tools to unstructured text. This typed information can be used to further categorize indexed documents, enrich assisted navigation topics available to the user, or to match items found in structured data, such as databases.

Named entities are words or phrases matching three types of specific data:

- proper nouns (for example, people, places, organizations, events),
- temporal expressions (for example, times, dates)
- number expressions (e.g., monetary amounts, percentages).

For example, some commonly used named entities include:

- People: Barack Obama, Madonna
- Locations: Paris, Trafalgar Square, Czech Republic
- Events: 2012 Olympic Games
- Dates: 1999/12/31, July 4th
- Addresses: 10 Downing Street, London, SW1A 2AA
- Monetary amounts: $14,000
The Exalead CloudView platform uses pattern-matching algorithms, over words and parts of speech, to detect named entities in indexed content. The platform includes ready-to-use named entity extractors for people’s names, geographic locations and company and organization names, both in English, French, and other languages. These extractors are defined by using the Exalead CloudView pattern matching technology, so that extractors for other kinds of names entities or other languages can be easily built on top of this module. The pattern matching algorithms involved in the detection of names entities must deal with many ambiguous patterns and performance issues. Machine learning techniques allow for newly discovered patterns which might be entities to be classified into their types, using the context found around the entities. Named entity filters combine multiple criteria to correctly detect the word patterns, including:

- **Thesaurus:**
  For example, “Reynolds” is known to be a last name, “New York” a location and “CNN” an organization,

- **Morpho-syntactic rules:**
  For example, a “$” followed by a number is a monetary amount,

- **Disambiguation rules:**
  Used, for example, to determine whether the “Reynolds” referred to in a document is an individual named “Reynolds” or the R.J. Reynolds Corporation.

The thesaurus and morpho-syntactic rules work in tandem, for example, recognizing as a proper name a “first name” (thesaurus) followed by a “capitalized word” (a morpho-syntactic rule recognized by a simple regular expression). The built-in thesaurus files shipped with the Exalead CloudView product have been constructed by automatically extracting comprehensive lists of first names, people names and companies from the Web. Nonetheless, named entities detection heuristics are by nature general, and it is possible that certain entities will not be detected. For example, an organization’s name that is not included in the dictionary will not be recognized.

![Figure 5 - Named Entity Example](image)

One use of recognizing names entities and related terms in Wikipedia articles in the entity-coded tag cloud appearing on the right of the screen. Wikipedia categories are explicitly noted in the articles. The query here was “sea turtle”.

![Figure 5 - Named Entity Example](image)
4.2 Contextual Entity Matching

To ensure the successful classification of the documents in a corpus and consequently the most relevant search results it is necessary to define multiple ontological rules and a complex disambiguation policy. This extension of named entity matching is referred to as Contextual Entity Matching (CEM). Using CEM you can match for an entity using several features of this entity to disambiguate. You can disambiguate using rules based on optional or mandatory entity matches and/or regular expressions.

For instance, when looking for a named entity, disambiguate Madonna (the singer) and Madonna (not the singer) because the other entity "Guy Ritchie" occurs in the context. Disambiguate "gone wind" to match to the entity Gone with the Wind (book title) using a regular expression that defines the distance between the words and the match and also because the other entity "Margaret Mitchell" occurs in the context.

This technology can be used to match for people in web pages or documents or to better comprehend user queries. Exalead CloudView defines default ontological rules for CEM. However, the user can intervene in the Mining-Of-Text Architecture to apply ontological rules that exploit knowledge bases and maximize the efficiency of entity matching and disambiguation. This can be particular useful when providing services such as yellow-page search.

4.3 Categorization

Categorization means analyzing the content of a document and using that content to choose one or more categories that the new document could belong to. Exalead CloudView recognizes metadata associated with a document, and can index it. This data sometimes includes customer specific categories.

Exalead CloudView includes modules that will also generate typed metadata for a wide range of objects:

- language of the document (English, French, German, ...)
- type of document (PDF, Word, e-mail, html)
- source of the document (local PC, blog, forum, commercial site, non commercial site)
- date the document was created
- media type (text, audio, video)
- presence of a face in an image
- primary color, size, orientation of an image

In addition to these explicit or structural categories, Exalead CloudView can also automatically categorize new documents into a hierarchical category scheme.

Exalead CloudView performs automatic categorization by learning from a set of already classified documents. All the documents from each category contribute to a model of that category. The words that are characteristic to that category are used as clues to place a new document in the right category\(^2\). Users can supply the hierarchy and the documents that are to be integrated into Exalead CloudView processing. By default, Exalead CloudView supplies a module than can categorize documents into a user supplied classification scheme. The user can supply the classification hierarchy and typical documents in each category. Publicly available schemes such as the Open Directory Project subject hierarchy (www.dmoz.com) can also be used. A hierarchy of categories is sometimes mistakenly called an ontology, but ontologies contain logical rules in addition to a hierarchy of categories. Exalead CloudView in its current version classifies into category hierarchies, but users can intervene in the Mining-Of-Text Architecture to apply ontological rules as well.

---

Categories correspond to the subject area of a piece of text. Whole documents can be assigned to categories. Named entities can be of several different types for example, people are often labeled by their occupation, geographical sites by their location.

![Image](image1.png)

**Figure 6 - Exalead CloudView** can extract relations between people. This interface to the Miiget Exalead tool shows the occupations of people connected to the central figure throughout the Web.

### 4.4 Clustering

When no hierarchal structure is given beforehand, it is still possible to find and group similar documents, or detect different versions of the same document. This semantic processing groups documents based on their content. When two documents contain very similar content, they can be considered as versions of each other, and be presented together sorted by modification time, or length. **Exalead CloudView** offers a clustering module which groups documents from pools of documents into similar groups. This module also offer the possibility of choosing the most representative document in the cluster, or the most recently added document, and to show this document, along with a more succinct list of similar documents. When the documents come from a feed, such as an RSS news feed, documents are queued and periodically (for example, once per day, or once per hour) the documents are reclustered into the most important stories of the day.

### 4.5 Sentiment Analysis

Sentiment analysis is the name given to extracting and quantifying positive and negative reactions from text. **Exalead CloudView** includes vocabulary analysis that detects positive and negative sentiment words. This level of semantic tagging can then be used as part of the decision making process in a business process when it is rendered visible.

Along with the informational side of text, there is an emotive side in some cases. Customer e-mail might include complaints or compliments. People freely express their emotions about a product or a company in online forums and blogs. Sentiment analysis is the name given to extracting and quantifying positive and negative reactions from text.
Text is analyzed by comparing each word to a list of positively oriented words (for example, fantastic, superb, good, excellent, clever,...) and to a list of negatively oriented words (for example, disgusting, dirty, stupid, disappointing, depressing,...).

The presence of negation (for example, not good) changes the orientation of a word. Some words may have a higher weight than others (e.g., hilarious is stronger than funny). A calculation involving the weights and orientation of the effect-bearing words determines whether a text passage is positive or negative. When an entity has been identified in the passage, it is affected by the overall tone. Sentiment analysis over large quantities of text is relatively easy, once the basic vocabulary has been marked with its semantic orientation.

Exalead CloudView includes vocabulary analysis that detects positive and negative sentiment words. This level of semantic tagging can then be used as part of the decision making process in a business process when it is rendered visible.

4.6 Event Extraction and Triples

Exalead CloudView allows a user to:

- associate additional metadata
- detect events
- produce Resource Description Framework (RDF) triples

for documents in the corpus based on patterns found during document semantic processing.

Entity extraction and categorization allow us to type information found in unstructured text. Low level linguistics allows us to recognize different forms of the same item, and to find syntactically defined structures (such as noun phrases). These two functions of typing and matching are the basis of semantic processing. They are brought together in the function of event detection. Who did what to whom? What happened where?

In the pipeline of semantic treatment of input documents, we can scan the annotations that have been provided by previous steps to add additional metadata.

For example, from the following sentence:

ABC Corporation today decided to purchase an approximate 3.41% equity interest

shallow parsing and entity extraction can extract, among other information, the following pattern

<enterprise>, purchase, interest

---

Exalead CloudView allows the possibility to a user, in a module called Query Set Matching, to associate additional metadata to documents in function of patterns found during document processing. Given the above analysis Exalead CloudView’s semantic processing could add the additional semantic metadata: event="acquisition". Combined with Exalead CloudView’s ability to recognize synonyms (for example, “purchase” ~ “buy”, “acquire”, “obtain”), this semantic matching can recognize a wide variety of different expressions as examples of the same event.

Exalead CloudView can also produce RDF triples from document contents. RDF stands for Resource Description Framework, which is a standard for storing semantic units in the Semantic Web. A triple consists of a subject, a predicate, and an object. One way of looking at this is that a subject is an entity, a predicate is an attribute type and the object is the value of the attribute. RDF triples are being used as bridges between unstructured text and semantic representations of knowledge4.

4.7 Business Item Extraction

Business Intelligence, traditionally, involves analyzing database data to correlate business facts in order to understand business processes and make decisions. These facts are the visible effects of business: sales, inventory, customer and product data. The database structure incorporates the business model of the facts important to a business. Unfortunately, this approach based upon exploiting structured databases leaves aside the information contained in unstructured sources, such as customer emails, competitor news releases, blogs, etc., where one can find the causes of what happens in business. Much of the semantic information mentioned in preceding sections can be extracted from these information sources, and be used to explore this unstructured information in a structured manner.

Business Items are normalized, structured items that can interact with the database world. In order to extract information and create business items for a specific business from raw data, there are three steps.

- Specify the sources of the unstructured data to be exploited. This could be a list of websites, e-mail repositories, shared file folders, etc.
- Specify the model of the business information to be extracted. Exalead has defined a language for defining the schema of items, based upon the Notation35 formalism that describes objects manipulable by the Semantic Web.
- Produce the factory elements that convert unstructured text into these business items. In Exalead CloudView, a factory element is a combination of the modules available in the Mining-Of-Text Architecture described above which process and transforms text into semantics. Many of the modules needed to fill in the values of most business items are already available in the Exalead CloudView package. Any specific modules can be integrated using Java or C#.


These three steps allow Exalead CloudView to collect and process business information from previously untapped sources. The conversion into business items allows the information to be joined with previous structured database items, providing new dimensions on which business decisions can be made, looking at both the causes and the effects of business activity.

CloudView also provides standard and modularizable interfaces for exposing and exploring both types of information.
5 Semantics for Search Presentation

Exalead CloudView is packaged with an extensive list of semantic processors that you can configure in the MOT. Each processor can enrich the document using all the information extracted by previous processors. The semantic processors available for query parsing and search presentation are:

- Approximate matching
- Phonetization
- Spell checking
- Synonyms matching

The search modules also use the semantic information generated during indexing:

- To perform navigational search based on related terms.
- To perform navigational search for categories based on extracted semantic information.
- To calculate the relevancy.

5.1 Linguistic Operators

Once data is marked up and stored, we can still find new ways to explore it. In addition to exploiting stored facets, Exalead CloudView offers a number of tools that allow the user, on the fly, to match different things, and to create more precise queries.

Exalead CloudView offers phonetic search, approximate word matching, truncation operators, excluding words including:

- **soundslike**: Phonetic search is language-specific and uses a rule-based system to transform a written form to a phonetic form (a form using symbols to represent the speech sounds of a language). For example, the phonetic form of the words “sun” and “son” is “[suhn]”. This system involves specialized processing for some languages, such as Pinyin for Chinese, and Romaji for Japanese, with two systems used to transliterate Chinese/Japanese words into occidental alphabets. When using phonetics-based word matching, Exalead CloudView uses language-specific dictionary to find words that have the same phonetic form as the query terms, for example, matching “exallead” and “exalead”. Search queries can then be executed against all the words found by the phonetic lookup, and the distance between the matched words and the query terms is used in the relevancy score computation to prefer the words which are closer to the terms entered by the user.

- **spellslike**: Approximate word matching is similar to the phonetic word matching described above. The Exalead search engine first uses syntactic rules (e.g., adding or removing characters, changing one character into another) to determine a set of words having a similar spelling as the query terms, as with the “exalaed”/“exalead” example). Then, the search query is executed against that list of words, taking into account the ‘distance’ between the approximated words and the query terms. The approximate search method is complementary to the spell checker described below, in the sense that it not only allows users to find correctly spelled documents with a misspelled query, but also to find misspelled documents with a correctly spelled query. For example, without approximate search, a misspelled document containing the word “Exalaed” could remain totally hidden to users searching for “Exalead”. This is not acceptable for applications like e-commerce or compliance, where an item being not found by a search query can have serious consequences.

- **Truncation**: The Exalead CloudView query module provides a truncation operator allowing users to search for words beginning with a given prefix. For example, the query “messag*” will find documents containing words such as “messaging” and “messages”. This is particularly useful when users are uncertain of a term’s spelling, or when they want to intentionally broaden their search to encompass many variants of a given words.
• **Regular expressions**: Going beyond simple truncation, **Exalead CloudView** also allows searching on arbitrary regular expressions. For example, in the search "/exa*lead/", the "*" indicates the preceding "a" can be repeated 0 or more times. This would return, for example, "exalead", "exaalead" and "exaaaalead". The option to search using regular expressions is particularly valued by power users such as document specialists. When processing a regular expression query, **Exalead CloudView** first retrieves a set of words matching the regular expression from the dictionary. For performance reasons, the retrieved list is limited to a [configurable] size, and in the case of a truncated list, the engine only keeps the most frequent words. Afterwards, the query is executed against the set of matching words.

- In addition to allowing users to search for variants on individual words using techniques like truncation and regular expressions, the Exalead system enables users to combine techniques to better match entire phrases. For example, a user may enter "/gr(a|e)y/ whale". This combines a regular expression "/gr(a|e)y/" plus a regular keyword (whale). The result set would include exact phrase matches for either "grey whale" or "gray whale".

• **Operators**: **Exalead CloudView** also provides another method for combining query terms using logical and positional operators to more precisely pinpoint results. The comprehensive set of query operators Exalead supports includes:
  - Boolean operators (AND, OR, NOT),
  - Word Position Operators (NEAR, BEFORE, AFTER, NEXT), and
  - Parenthesized Expressions (fast OR speed) AND NOT light).

• **Synonym expansion**: **Exalead CloudView** also offers the possibility to add in user defined synonym lists that are used to expand queries. For example, this list might contain the terms DB, database, and "data base" as synonyms. When a user enters one of these terms in a query, **Exalead CloudView** would add the other two (with the OR operator).

5.2 **Stemming**

Once a token has been recognized, we may have to further break it down into meaningful subparts. After the words are tokenized and segmented, we apply a further step in order to map the way words appear in text to a normalized dictionary entry using stemming or morphological analysis.

This mapping can be through a relatively simple process known as stemming. Stemming involves recognizing the root of a word and pruning off any suffixes (and some prefixes). Such algorithms have been used in computer treatment of text for forty years. For example, common suffixes for verbs in English are 
- *s*, *-ed*, and *-ing*. **Exalead CloudView** integrates stemming and morphological analysis for many languages. Morphological analysis means recognizing not just suffixes, but different forms of the same word (think, thought, for example). Morphological analysis requires large dictionaries for each language, and a method for mapping variant words forms to their dictionary entry. **Exalead CloudView** provides morphological analysis for most of the major European and Asian languages, and simpler stemmers for other languages. The search example below illustrates using one word form "cloning" and retrieving

---


documents with different forms of the same word “cloned” and “clone”. This low level natural language processing can be achieved using either stemming technology or more complicated and more accurate morphological analysis. Exalead CloudView offers both approaches depending on the language.

5.3 Spell Checking

When Exalead CloudView is used to query documents, it can calculate the terms that are found often in documents relevant to the query.

While processing a corpus of text, Exalead CloudView gathers statistics about word use for each language. It stores the frequency of words, the sequences of words found around it, the variant forms that are mapped to the same form, and the regular expressions that correspond to the word. Some of these statistics, such as the frequency of a word, enter into the calculation of document relevance during search. A rare word appearing in a query and in a document means they are a better match. Other pieces of information are useful for spelling correction suggestions.

Exalead’s patented “Search by Serendipity”™ produces dynamic, guided navigation of results based on existing and extracted metadata. This navigation assistance, together with user aids like phonetic and approximate spelling search and auto-spelling correction, fosters more natural, more successful search.
5.4 Related Terms

All the sequences of words found during the indexing of a corpus of text are collected to create large dictionaries of terms with their frequencies in all the documents. The most commonly occurring terms in the corpus are associated with each document at indexing time. When Exalead CloudView is used to query documents, it can calculate the terms that are found often in documents relevant to the query. In a search interface using Exalead CloudView, the most salient terms can then be displayed in a “Related Terms” window that can be used to refine the initial query.

![Diagram of Dictionary Update and Extraction](image)

5.5 Summarization

A summary is a shortened form of a document, or an extract from a document that contains a portion of the semantically interesting part of the document. Summaries can be generated by analyzing the vocabulary of a document, selecting the words and phrases that are most frequent in the document, then going back to select the sentences which contain the greatest density of these frequent phrases. The extracted sentences are presented in the same order as in the original text and produce short summary of the document.

---


When documents are retrieved with respect to a query, most search engines produce a directed summary; centered around the words of the query. These summaries, displayed in the list of results, are often called *snippets*. Summaries can also be centered around entities, dates, places, etc.

![Image of search results snippets](image)

**Figure 11 - Snippets from the search results**

These are short extracts of the pertinent part of a document with respect to a query.

### 5.6 Question Answering

Information access covers both the functions of querying and of exploring structure found in or extracted from information. Sometimes, though, we are just looking for a short answer to a question. Question answering is part of semantic processing that attempts to find the shortest answer to a given question. Question answering systems take as input a user query expressed as a normal question, for example, “What river runs through Moscow?” or “Where did Genghis Khan die?”. These factual questions are treated by first deciding what type of information the user is looking for (a place, a data, a quantity, a person). The other words in the query are used to find relevant passages in the corpus. Finally named entity extraction (see above) is used on these passages to find the new information that the user seeks. If the corpus is big enough, and the information is found in many places (as on the Web), then there might be many responses to the query, and the entity appearing the most often is usually presented as the answer to the question.

![Diagram of question answering processing pipeline](image)

**Figure 12 - Question answering processing pipeline**
5.7 Faceted Analysis

Information access is much more than ranked lists of search results. As semantic processing extracts richer and richer meaning from both structured and unstructured data, one popular way of exploring information is faceted search. A facet is a clearly defined property or characteristic of a class or a subject. In the 1930s, the Indian mathematician and librarian Shiyali R. Ranganathan proposed faceted search as an alternative to the rigid structure of systems such as the Dewey Decimal System\(^\text{10}\). The presence of facets as an additional tool is becoming more and more frequent as a tool for information access. For example, the Exalead.com/search page shows a side bar that allows the user to refine their initial query using a number of facets, covering the type of document, the source of the documents, the language, specific categories, and many other types of facets corresponding to metadata either present or automatically associated with the original documents by Exalead CloudView. In this interface, facets define

---

dimensions of interest that can be reinforced or eliminated to refine query results and meaningfully explore the space of answers. Facets are popular search tools on shopping sites, allowing the user to specify, price, sizes, colors, and other areas of interest.

Figure 14 - The “Narrow your Search” window shows faceted analysis
6 CloudView Semantic Features

Exalead CloudView combines the best of natural language processing techniques and web technologies to provide a software platform that exploits all your business information, both structured and unstructured. The Mining-of-Text open architecture provides total flexibility to ensure that your business application combines the optimal linguistic features in an optimal sequence. Exalead CloudView provides dynamic and static linguistic resources used by the following totally configurable processors:

- natural language processors
- semantic processors

6.1 Natural Language Processors

- **language detection**
  Exalead CloudView supports *language detection* which allows the natural language processors to perform tokenization, normalization and morphological analysis correctly. It currently recognizes the principal 54 different languages used on the Internet.

- **tokenization and normalization**
  Exalead CloudView supports *tokenization and normalization* for all major languages but also provides an API to customize the tokenization and normalization for any language.

- **lemmatization, stemming**
  Exalead CloudView supports *lemmatization* and integrates the language model for possible tags and word sequences for the major world languages. The *stemmer* supports 15 major languages (da, de, en, es, fi, fr, hu, it, nl, no, pt, ru, sv, tr, ro).

- **part-of-speech analysis**
  Exalead CloudView supports part of speech analysis that can perform disambiguation based on a learned data set.

- **spell checking**
  While processing a corpus of text, Exalead CloudView creates word and word co-occurrences dictionary. These dictionaries can be used for *spell checking* to improve the efficiency of search.

- **linguistic resources**
  While processing a corpus of text, Exalead CloudView gathers statistics about word use for each language. Thus your *linguistic resources* are tailored to your business data to improve the efficiency of your application performance, such as search.

6.2 Semantic processors

- **named entity extractors**
  Exalead CloudView includes ready-to-use *named entity extractors* for people’s names, geographic locations and company and organization names. Extractors for other kinds of names entities or other languages can be easily built on top of this module.

- **ontology matching**
  While processing a corpus of text, Exalead CloudView supports the extraction of entities or concepts from unstructured data based on an ontology. For example you can extract the list of company employees and services in all documents.

- **context entity matching**
  Using context entity matching you can match for an entity using several features of this entity to disambiguate.

- **fuzzy ontology matching**
  Furthermore you can perform fuzzy ontology matching on text by allowing the usage of boolean operators between concepts. For example the *Bill & Melissa Gates Foundation* ontology element will also match text with *Bill Gates Foundation* and *Gates Foundation*. When you search for this organization, we are able to give all different matches ranked by similarity with original form.
• related terms extraction and detection
  While processing a corpus of text, Exalead CloudView extracts related terms from documents, detects related terms and associates the data with the document. This data can be used to refine the initial query during search.

• categorization
  Exalead CloudView performs automatic categorization, based on a user defined hierarchy, by learning from a set of already classified documents.

• clustering
  When no hierarchal structure is given beforehand, it is still possible to find and group similar documents, or detect different versions of the same document.

• event detection
  Entity extraction and categorization allow us to type information found in unstructured text. Low level linguistics allows us to recognize different forms of the same item, and to find syntactically defined structures (such as noun phrases). Exalead CloudView brings them together to perform event detection. *Who did what to whom? What happened where?*

• faceted analysis
  Exalead CloudView supports faceted analysis. Users can refine their initial query using a number of facets, covering the type of document, the source of the documents, the language, specific categories, and many other types of facets.

• phonetization
  While processing a query, Exalead CloudView performs a phonetization of each token based on a set of phonetic rules (for example ‘vouature’ matches ‘voiture’ in french) to improve the efficiency of your business applications.

• approximation
  While processing a query, Exalead CloudView performs an approximation of each token based on a set of approximation rules (for example ‘colour’ matches ‘color’) to improve the efficiency of your business applications.
<table>
<thead>
<tr>
<th>Glossary</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crawler</td>
<td>A web crawler (also known as a web spider or web robot) is a program or automated script that browses the World Wide Web in a methodical, automated manner.</td>
<td><a href="http://en.wikipedia.org/wiki/Web_crawler">http://en.wikipedia.org/wiki/Web_crawler</a></td>
</tr>
<tr>
<td>HTML</td>
<td>HTML, the initials of HyperText Markup Language, is the predominant markup language for Web pages.</td>
<td><a href="http://en.wikipedia.org/wiki/HTML">http://en.wikipedia.org/wiki/HTML</a></td>
</tr>
<tr>
<td>Information Retrieval</td>
<td>Information retrieval (IR) is the science of searching for documents, for information within documents and for metadata about documents, as well as that of searching relational databases and the World Wide Web.</td>
<td><a href="http://en.wikipedia.org/wiki/Information_Retrieval">http://en.wikipedia.org/wiki/Information_Retrieval</a></td>
</tr>
<tr>
<td>JavaScript</td>
<td>JavaScript is a scripting language most often used for client-side web development.</td>
<td><a href="http://en.wikipedia.org/wiki/JavaScript">http://en.wikipedia.org/wiki/JavaScript</a></td>
</tr>
<tr>
<td>Natural Language Processing</td>
<td>Natural language processing (NLP) is a field of computer science concerned with the interactions between computers and human (natural) languages. It is often considered a sub-field of artificial intelligence.</td>
<td><a href="http://en.wikipedia.org/wiki/Natural_Language_Processing">http://en.wikipedia.org/wiki/Natural_Language_Processing</a></td>
</tr>
<tr>
<td>Named Entity Recognition</td>
<td>Named entity recognition (NER) is a subtask of information extraction that seeks to locate and classify atomic elements in text into predefined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc.</td>
<td><a href="http://en.wikipedia.org/wiki/Named_entity_recognition">http://en.wikipedia.org/wiki/Named_entity_recognition</a></td>
</tr>
<tr>
<td>PHP</td>
<td>PHP is a scripting language originally designed for producing dynamic web pages.</td>
<td><a href="http://en.wikipedia.org/wiki/PHP">http://en.wikipedia.org/wiki/PHP</a></td>
</tr>
<tr>
<td>Regular Expressions</td>
<td>In computing, regular expressions provide a concise and flexible means for identifying strings of text of interest, such as particular characters, words, or patterns of characters.</td>
<td><a href="http://en.wikipedia.org/wiki/Regular_Expression">http://en.wikipedia.org/wiki/Regular_Expression</a></td>
</tr>
<tr>
<td>Semantics</td>
<td>Computational semantics is the study of how to automate the process of constructing and reasoning with meaning representations of natural language expressions.</td>
<td><a href="http://en.wikipedia.org/wiki/Computational_semantics">http://en.wikipedia.org/wiki/Computational_semantics</a></td>
</tr>
<tr>
<td>Taxonomy</td>
<td>A hierarchical representation of terms or concepts, where a concept lower in the hierarchy is a more specific instance of a concept found above it.</td>
<td><a href="http://en.wikipedia.org/wiki/Taxonomy">http://en.wikipedia.org/wiki/Taxonomy</a></td>
</tr>
<tr>
<td>Thesaurus</td>
<td>In information technology, a thesaurus (also known as Ontology) represents a database or list of semantically orthogonal topical search keys, with possibly explicit relations between the terms.</td>
<td><a href="http://en.wikipedia.org/wiki/Thesaurus">http://en.wikipedia.org/wiki/Thesaurus</a></td>
</tr>
</tbody>
</table>
About Exalead

Founded in 2000 by search engine pioneers, Exalead is a global software provider in the enterprise and Web search markets. More than 170 companies worldwide and 100 million unique users a month rely on Exalead’s information access platform to search, discover, and manage their information assets for faster, smarter decision-making, real-time unified data access, and improved productivity.

Exalead’s team includes industry-leading experts in information search, non-structured data analysis, and natural language processing. This team has concentrated its R&D efforts on meeting its clients’ need to collect, transform, index, and search arbitrarily complex data from heterogeneous sources.

As a result, the Exalead CloudView product has emerged as a uniquely successful platform for automatically structuring very large volumes of nonstructured data, such as email messages, Office documents, presentations, Web pages, blogs, forums, and RSS feeds.

CloudView is currently being deployed for:

• Enterprise Search
• Extended Business Applications [EBI, Smart CRM, Intelligent Compliance, etc.]
• eBusiness (search and content enhancement for high traffic websites)
• Data Management (database offloading, data migration and information lifecycle management [ILM])
• Embedded Search for OEMs/ISVs

For more information, please visit http://www.exalead.com/software. The company’s public WWW search engine is accessible at http://www.exalead.com/search.