



STOP SEARCHING  
COGNIZER AI  
START KNOWING

COGNIZER CORPORATE BRAIN PLATFORM  
Technical White Paper



## Introduction

In today's high-performance enterprises, data is king. Many experts say that, "Data is the new oil!" It is how top companies execute, compete and dominate their industry. But in most companies, more than 90 percent of all data is unstructured. It is in documents, emails, meeting transcriptions and collaboration systems like Slack. What if we could turn this content into a data source that could be queried like a database? What if we had a Corporate Brain that knew every fact in the enterprise, and it could be even more valuable to us than our core enterprise systems?

Well, unfortunately, that is not the world we currently live in. In today's world, if you ask an enterprise knowledge worker if she is overwhelmed with the amount of information coming at her in the form of documents, emails, calls, and meetings, the reply will almost certainly be that she suffered from this in spades. But if you ask the same knowledge worker if she has all of the information she needs to do her job and be effective, she will shake her head hopelessly. What an odd duality! Way too much information, but not nearly enough knowledge and insight to do your job.

Enterprises are trying to solve this problem with Enterprise Search. But the value proposition of Enterprise Search is a downward-facing exponential curve. The more information you give it, the more results come back in a search. Then, as a user, you get to go through the results and try to figure out where the real answer is in mounds of information returned. Some search engines try to solve this "needle in the haystack issue" by organizing the content by topic or keywords. But the answer to the haystack problem is not color coding the hay, it is providing a powerful magnet to find the needle.

The solution to this problem is a Corporate Brain! A Brain, whether it be a Corporate Brain or a human brain, solves the problem in an entirely different way. Brains take in information, process the information into distinctive facts and then store those facts in its memory. For the human, our memory is our cerebral cortex. For the Corporate Brain, it is a powerful and massive Fact Graph Database.

As humans, our memory is scarce and unreliable. But to an enterprise, memory in the form of a Corporate Brain is nearly unlimited, highly reliable and can provide incredible value. It is essentially "Jarvis" for your company. It is a "thought partner" that can read everything, remember anything, and can create boundless knowledge and insight.

At Cognizer, we invented the Corporate Brain. In 2020, we are about to unleash one of the most powerful enterprise tools ever discovered. It is time to give your company a massive boost in IQ!



## The First Corporate Brain: Cognizer

Cognizer is the world's first Corporate Brain. It is based on a new technology that we call Natural Language Intelligence (NLI), and it establishes intelligence very similar to the way that humans do. When humans are considered "intelligent," they are very good at identifying and inferring important facts from information. They also know a large quantity of facts and can retrieve them very quickly. Cognizer's NLI is very similar, and is based on a massive Fact AI, paired with a powerful Inference AI.

After spending 2 years and \$2.5M in R&D, the Cognizer Corporate Brain will be commercially available in early Q1 of 2020. Over the next 3 years, we will invest an additional \$10 million in this powerful platform.

### Enterprise Facts and Global Facts

To become intelligent, Cognizer must read and understand as many facts as possible. It does this in three ways. First, Cognizer has completely read all of Wikipedia, DBpedia, Freebase, and the CIA World Factbook. It comes pre-populated with over 1 billion Named Entities and 35 billion Facts about those Named Entities.

Second, Cognizer is directly connected to Wolfram Alpha and can do Ph.D. level math instantly. Wolfram Alpha also gives Cognizer specialized information about medicine, chemistry, engineering, money, politics, and entertainment.

Finally, Cognizer can read content within the enterprise, like documents, emails, collaborations, and enterprise data. Each sentence of content is digested by a sophisticated UIMA Pipeline that includes over 15 NLP Processes and 10 deep learning models. The results are pristine facts that are stored in what we call the Hypergraph.

### Harvesting Facts

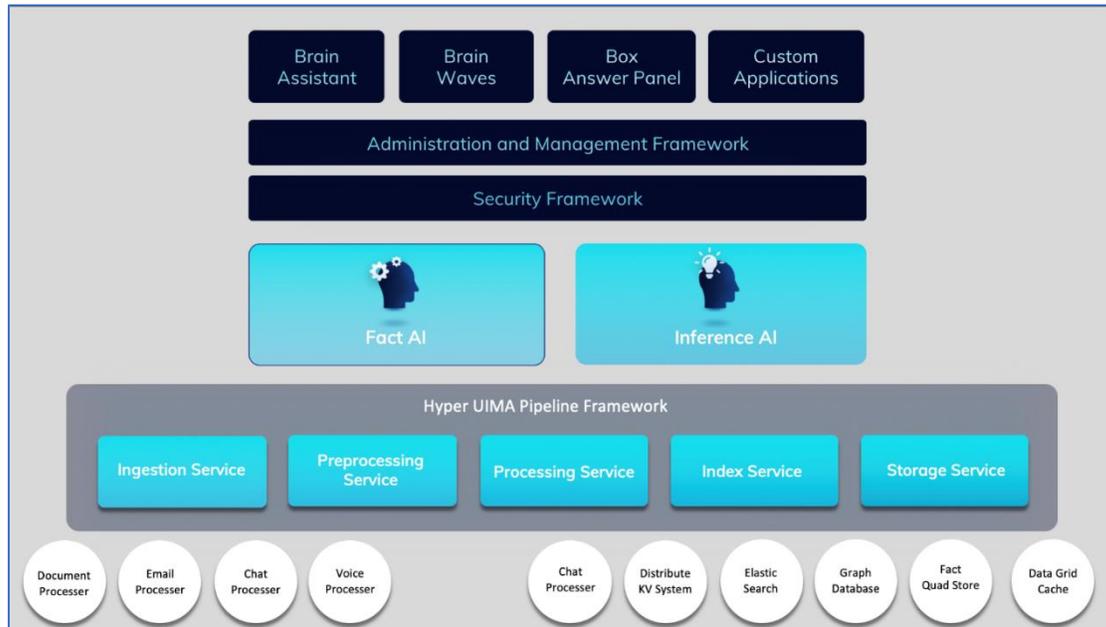
As Cognizer reads this important information, it is looking for Named Entities. Named Entities are people, companies, locations, technologies, topics, etc. Essentially, anything that is a "thing". It then looks for any facts about those Named Entities. It collects these facts and then saves them in a highly scalable and distributed "Triple Store" database called Cassandra.

Facts are collected throughout each document, across documents, and even across repositories (i.e. Box Documents and Exchange Email). When reading all the repositories of an enterprise, all of the facts about a subject (Named Entity) are collected and saved and immediately made available to the user.



## The Cognizer Brain Platform

Like the human brain, the Cognizer Brain has two halves. The left hemisphere is a Fact Artificial intelligence (AI) and the right hemisphere is an Inference AI.



The two brains not only work parallel with each other; they are constantly enhancing each other through the Learning Optimizer.

- If the answer to the posted question resides in the Fact Graph, the Brain's Fact AI answers. Thereafter, both the posted question and delivered answer is sent to the Inference AI for further training.
- Alternatively, when the answer is not in the Fact Graph, the Inference AI answers the question through semantic inferencing. The resultant answer is then sent to the Fact AI to harvest additional facts into the Fact Graph.

The system is constantly learning, and because the Fact AI and the Inference AI work on two completely different paradigms (one on discreet facts and one on inferred facts), they are constantly optimizing each other.

Here is a simple example. Cognizer reads content that says, "Jack Porter's wife is Susie, and their children's names are Becky and Katie." Since the sentence does not specifically

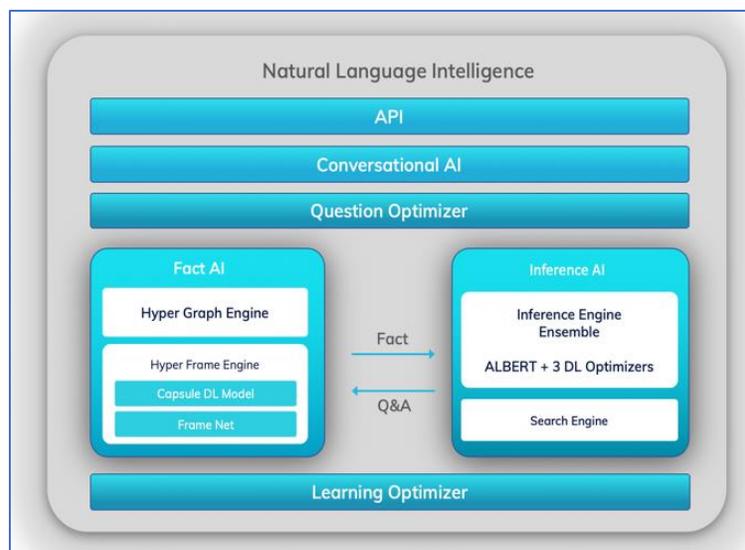


say who Susie's husband is or who Becky's father is, this information must be inferred. When a question like this is asked, the Fact AI looks in the graph for the information and if it is not there. It then queries the graph for information it does know about Jack and gives it to the Inference AI. The Inference AI makes the inferences, establishes several more facts about Jack and gives them back to the Fact AI to store in the Hypergraph. The Inference AI has now been optimized by training on this information, and the Fact AI has been optimized by the inferences of the Inference AI.

### Question Optimizer

When a user asks a question, it is converted to text and then sent to the Question Optimizer. Since users cannot be expected to ask the question in a manner that the AIs will best understand, this service will do it for them.

Users might simply put in keywords, expecting a search. They might put a compound sentence, use slang or an acronym, or they might make a spelling or grammar mistake. No matter what is provided, the Question Optimizer will make it a proper, well-formed sentence. Then the question is optimized for the AIs. In fact, it optimizes the question twice, once for the Fact AI and once for the Inference AI. Then the question is sent to the AIs for processing.



### Fact AI

The Fact AI is based on a powerful AI technology called a Hypergraph. The Hypergraph uses frames to understand the lexicon of a sentence and then creates facts from those frames. Cognizer uses an AI-enhanced version of the FrameNet. FrameNet was developed at International Computer Science Institute at Berkeley, CA. Cognizer has



enhanced FrameNet by adding a powerful deep learning capsule network that not only optimizes each of the frames, but also creates new frames as additional information is consumed and the lexicon is expanded.

The Cognizer Hypergraph is based on a powerful triple-store database that can be materialized as either a hierarchical graph or in time-series dimensions. This database can be horizontally scaled to support trillions of facts with a subsecond response time.

### [Inference AI](#)

The Inference AI is based on a powerful Conversational AI. Cognizer uses an AI ensemble here that includes Google's ALBERT, which has been enhanced by three other deep learning models for optimization. By itself, ALBERT achieves state-of-the-art results that exceed human performance, and it has claimed first place on the leaderboard of all three of the NLP Grand Challenges, including SQuAD 2.0, GLUE, and RACE.

The Inference AI leverages Bidirectional Encoder Representations from Transformers (BERT), and has been enhanced with additional deep learning models for both question optimization and learning optimization.

### [Enterprise Content](#)

To support the enterprise, Cognizer can digest virtually any kind of business content. It reads each piece of information, sentence by sentence, word by word, and harvests business critical facts that, to date, have really just been information exhaust. It is like taking your most precious treasure and burying it in the backyard. The value is still there, it is just not being utilized. Cognizer basically turns your content into a virtual database that can be queried just like any of your important enterprise systems.

The Cognizer Platform has connectors that allow it to ingest this important information. Each connector is optimized for the specific type of content, and the connector interface has been published to allow new connectors to be developed by third-party developers.

### [Document Connector](#)

Cognizer has developed a tightly coupled connector to the Box Cloud Content System. This solution includes deep integration with the Box's profiles and attributes, security, and workflow. Cognizer provides single sign-on with Box and is tightly integrated with Box's access and authorization.

Cognizer fully integrates with Apache Tika and can read content in over 1,000 formats. Special programming has been developed to read very sophisticated tables within a document and aggregate important data across documents.



In 2020, Cognizer will be developing Document Connectors to Google Drive, Microsoft OneDrive, Network File Systems (NFS and IFS), and WebDAV.

### Email Connector

Cognizer supports the IMAP interface for reading email, and the connector has been optimized for both Gmail and Microsoft Exchange. The Email Connector reads the attribute data like To, From, CC, BCC, and Subject, as well as the body and the attachments.

An Email Body Processor also reads for specific information like a task request, contact introduction, meeting invitation, as well as special items like websites to review, videos to watch, books to read, and movies to see.

### Collaboration Connector

In addition to the content within Box and Email, in early 2020, Cognizer is developing a deep integration with Slack. This will include a Bot interface for asking Cognizer questions within the Slack UI, as well as reading Messages and Channels within Slack for content and facts.

Additional collaboration systems like Microsoft Teams and Google Talk will be developed in future releases.

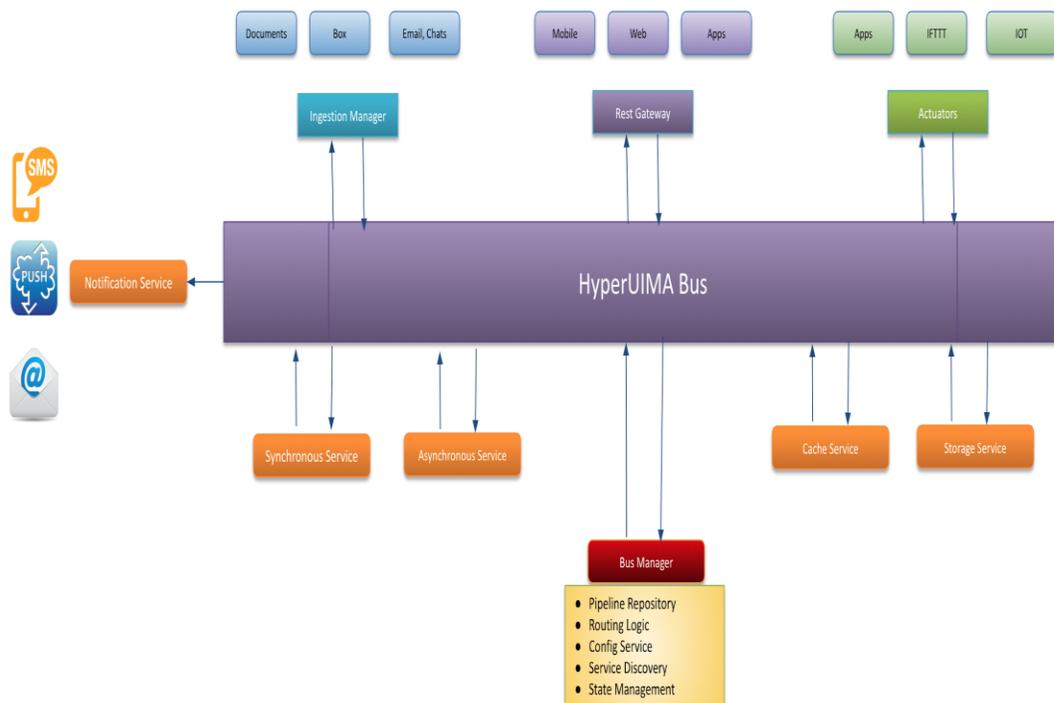
### Data Connector

In early 2020, Cognizer will release three Connectors for enterprise data. The first is a very simple REST/JSON interface. The services provides a unique URL. When a JSON payload is sent to the URL, it triggers Cognizer to consume the data. This allows the data to be consumed instantly and reduces the load on the servers so they will not be queried for data every 5 or 15 minutes.

In addition, Cognizer will be developing a deep integration into Zapier and the Robotic Process Automation (RPA) system Automation Anywhere.

### Hyper UIMA Platform

To read all of a company's business critical information, a system must be massively scalable. In many large corporations, this will be billions or trillions of documents and emails. That means a Corporate Brain like Cognizer must be built at Web Scale. It must be both vertically and horizontally scalable, expandable and extensible, secure, and very easy to manage.



The Cognizer platform is based on a Hyper UIMA Framework. Unstructured Information Management Architecture (UIMA) is an older framework originally developed by IBM more than 20 years ago and was used in IBM's Watson. The concepts of UIMA are very good, and it is utilized in many development projects. It allows an organization to analyze large volumes of unstructured information in order to discover knowledge that is relevant to a user. But the implementations that were used in UIMA are not nearly as scalable and reliable as modern web-scale architectures. Traditional UIMA introduces many challenges—including complex mechanisms—to attain interoperability, seamless scaling, interface flexibility, dynamic pipelines and reactivity.

HyperUIMA is very similar to the traditional UIMA paradigm (in terms of the message contract and format) and backward compatible with its interfaces. The difference being that HyperUIMA uses a reactive event-driven architecture, which has the following benefits:

- Enables faster end-to-end processing as services can be executed in parallel/asynchronously.
- Individual services can be scaled independently and allows for scaling the platform where the scaling is required the most. In addition, the availability of these services is guaranteed.



- Easier to add/update services as they can be plugged in/out of the event stream easily.
- Aligns well with an agile delivery model as teams can focus on particular services instead of the entire application.
- Control is distributed, so there is no longer a single orchestrator serving as a central point of failure.

Several patterns can be used with a reactive architecture to provide additional benefits. For example, Event Sourcing is when the Event Stream stores all of the events and enables event replay. This way, if a service went down while events were still being produced, when it came back online, it could replay those events to catch back up. Also, Command Query Responsibility Segregation (CQRS) can be applied to separate out the read and write activities. This enables each of these to be scaled independently. This comes in handy if you have an application that is read-heavy and light on writes or vice versa.

## HyperUIMA Architecture Overview

HyperUIMA is an abstracted event bus framework that supports both synchronous and asynchronous processing paradigms. All components plug into the event bus and can be individually scaled out. This allows for a fully fault-tolerant, durable and highly available platform.

### HyperUIMA Bus

The backbone communication bus can be implemented either over Kafka or using a reactive framework event bus (like Vert.x). The Bus manager is the central component in the HyperUIMA framework that defines and manages the overall operations of the Cognizer platform.

The following are responsibilities of the bus manager:

- Pipeline Repository — This subcomponent stores and serves the processing pipeline information for all functional flows within Cognizer, like the fact generation flow, ingestion flow, query processing flow, etc.
- Routing Logic — This subcomponent holds the rules or routing logic for the processing engines. Based on the inbound data, the routing logic is dynamically determined, and this service is leveraged by all the components on the bus.



- **Config Service** — All the global and component specific operational configurations are maintained in this subcomponent. All the subcomponents plugged into the bus leverage this service to store and retrieve configurations on the fly.
- **Service Discovery** — During bootup, all Cognizer components register themselves with the service discovery component, so that all the other services can discover the contracts for a given component, which can be either queue details or http endpoint details.
- **State Management** — Every component, at some point, needs to persist its state to effectively recover from failures. This subcomponent provides api's for all the Cognizer components to persist their respective states in a timely manner. During bootup, these components can restore their state before failure and resume from the point of failure.

### Pre-Processing

Once the information has been ingested, it is pre-processed. Most NLP systems do not do this, and they fail because they try to process a sentence that is not well-formed. Rather than try and figure out a sentence that is not linguistically correct, Cognizer pre-processes the sentence to make it well-formed.

The pre-processing begins by expanding all abbreviations. Using its powerful Fact Base, Cognizer looks up the abbreviation in the context it is being used and then inserts the proper abbreviated words. Next, all spelling and grammar is corrected and things like emojis, slang and urbanized words are replaced. Then, sentence boundaries are identified, and all run-on sentences are broken down.

This is the set of Pre-Processing Modules:

- **Spell/Orthography Checker**  
Cognizer's spell-checker incorporates a context sensitive algorithm that infers the right spelling for any misspelled word (wrong spellings and wrong words have become very common with different unstructured sources and autocomplete programs), The spell-checker is a hybrid of multiple algorithms (JamSpell, Hunspell, etc.) where the confidence scores generated by the individual algorithms are used to disambiguate the right word replacements.

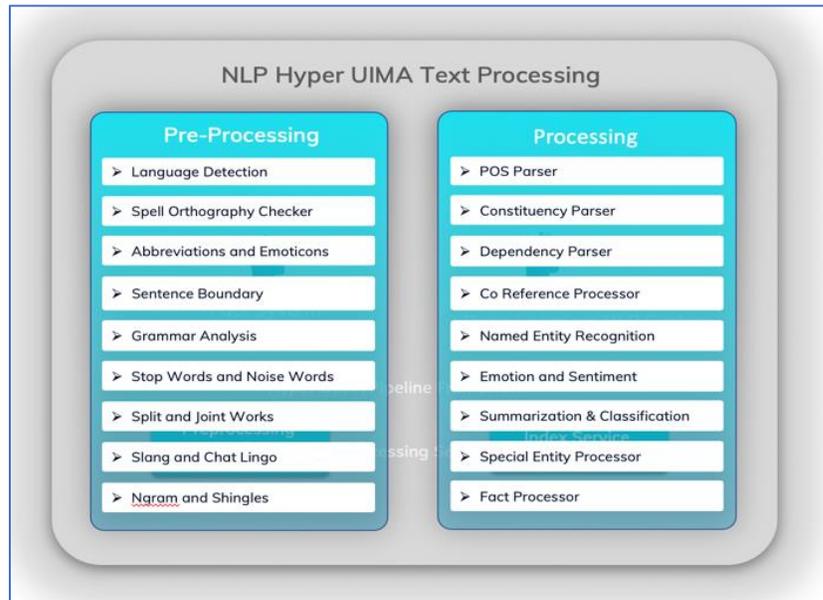


- **Language Detection**  
The language detection engine can identify major languages across the globe. This is very essential to activate the right language engine for processing the inbound text.
- **Abbreviations and Emoticons**  
Documents, chats, emails and conversations these days include a lot of short forms (P.S. IMHO, AFAIK, BRB, etc.), abbreviations (NFL, LOTR, ESPN etc.), and emoticons ( ;), ;=) ;0 ), With these words, it poses a huge challenge during downstream processing, and these words need to be transformed to their actual forms before any processing can be carried out.
- **Sentence Boundary**  
Sentences form the fundamental unit of work for all the processing engines within Cognizer. The Cognizer boundary detection algorithm marks syntactic boundaries of sentences within large chunks of text. Due to the inherent nature of text from sources like chats, emails and transcribed conversations, this process is based on several very sophisticated deep learning models.
- **Grammar Analysis**  
Cognizer grammar analyzer does a syntactic analysis of sentences to check if the text complies with proper use of the language grammar. It then corrects the grammar by rewording and adding/removing words and punctuations. This is an essential step that helps improve downstream processing accuracy.
- **Stopwords and Noisewords**  
As Cognizer scrapes knowledge from various raw sources, there is a chance that the text includes significant amounts of words either without any semantic or syntactic relevance, or words that are abusive in nature. These words are filtered by the Cognizer Noisewords Filter Engine.
- **Split and Joint words**  
Due to typing defects, autofill, autocorrect, autocomplete, and parsing failures from complex document format, words may either get split or joined. Cognizer's Word Analyzer detects these malformed words, and either splits them or joins them to form proper words in a given context.



- Ngrams and Shingles

Detecting multi-word entries is a huge challenge even today. Due to a mix-up in nouns, verbs and adjectives, our Ngram and Shingles Generator is backed by one of the largest entity dictionaries out there, which can recognize ngrams in virtually every context.



### HyperFraming

Once the pre-processing is complete and the sentence is well-formed, it is passed into HyperFrame Engine (HFE). One of the key reasons the Cognizer platform is so accurate is because of this engine. The HFE is a deep learning enhanced FrameNet algorithm. FrameNet was originally developed at the International Computer Science Institute in Berkeley, California and is based on a theory of meaning called frame semantics. A semantic frame can be thought of as a conceptual structure for describing an event, relationship or objects in the structure. As an example, if you book a reservation, make a reservation or secure a reservation, all of these words are in the same conceptual frame. Whereas if you read a book, it's the same word but in a totally different conceptual frame.

FrameNet by itself is very powerful, but when used in conjunction with an advanced deep learning model, it is a game changer. Cognizer's proprietary HyperFraming technology leverages deep learning to enhance frames, learn new frames and optimize the processing within frames. HyperFraming will set a new bar for high performance Natural Language Processing.



## Fact Processing

- **Sentencization** — This stage involves breaking large text segments into proper sentence boundaries.
- **Constituency Parsing** — This breaks the sentence into a tree and identifies its syntactical structure. This process identifies the parts of speech and how each part relates to the others. Constituency Parsing is used by other, more advanced algorithms in the platform to interpret the sentence and optimize the system's performance.
- **Dependency Parsing** — The fundamental notion of dependency is based on the idea that the syntactic structure of a sentence consists of binary asymmetrical relations between the words of the sentence. Dependency parser connects words according to their relationships. With Dependency Parsing, a sentence provides a grammatical structure and defines the relationships between the "head" words and the words that modify those heads.
- **CoReference Resolution** — The goal of CoReference Resolution is to determine which mentions in a section of text refer to the same real-world entity, property or situation. CoReferencing occurs when the same person or thing is represented with different words. This is often the case when using words like "his," "their" and "our." For example, "Joe went to the market with his dog."
- **Named Entity Recognition** — Named Entity Recognition locates all "things" in the document, including people, companies, locations, technologies, topics, etc. It then segments the entities into specific classes or categories.
- **Fact Materialization** — Once the Named Entities are identified, all facts about the entities are identified and organized as a tree graph with branches organized as a taxonomy. The facts are stored as a key value store and organized in a high-performance distributed graph.
- **HyperGraph** — The HyperGraph is a distributed AI enabled fact graph database that organizes the Cognizer Named Entities and facts into a powerful ontology. The nodes on the HyperGraph for similar categories (people, companies, locations) are schema-defined and organized to provide high-performance search and aggregation.

Cognizer actually has three HyperGraph of similar and complementary data



systems. The Global Fact Graph has public information from Wikipedia, Wikidata, DBpedia, CIA World Factbook and Freebase. The Enterprise Fact Graph has information from document stores like Box, Dropbox, network file shares, collaboration systems like Slack, and the Intranet; as well as enterprise information from systems like CRM, HRMS, ERP, and Service Management Systems. Finally, the Personal Fact Graph has personal data, such as email, contacts, tasks, meetings, etc. A user can render facts from each HyperGraph independently, or from all of the graphs at once.

- Indexing and Performance Engineering — Graphs are extremely fast at reaching specific node information, as well as aggregating information across nodes. The actual data for the graph is contained in a Key Value store, and the graph is materialized on top of this store. This is a much faster and scalable solution than traditional relational databases, and does not require joins, normalizations or specifically defined indexes.
- API and SDK — In addition to providing core applications, like Brain Assistant and Brain Waves, the Cognizer platform provides a full Application Programmable Interface (API) and a Software Developer Kit (SDK). The API is a set of micro-services that are implemented using REST and JSON.

The entire platform is containerized using Docker and can be managed using Kubernetes and Ansible. It is elastic, highly scalable and highly reliable. The whole design has been developed using today's most advanced technology for a web-scale solution.

## Enterprise Services

What makes Cognizer's Corporate Brain "Corporate" is that it is designed to the standards of today's modern enterprise. It possesses the 7 "-ilities" that strong internet technology (IT) teams demand. It has usability, maintainability, scalability, availability, extensibility, security and portability. A workgroup solution cannot scale to an enterprise answer. It will ultimately fail in every one of these critical areas. The Cognizer Corporate Brain platform is designed from the ground up to perform reliably to each of these business-mandated requirements.

## Enterprise Performance

Today, companies can't just operate at enterprise scale; they have to operate at Internet scale. Cognizer was designed to provide access to millions of customers, thousands of employees and hundreds of partners. Some companies at peak scale can hit millions of simultaneous users and require subsecond response times.



Cloud computing is the answer here, and clouds now can not only provide elastic scalability but also worldwide reliability and redundancy. Cognizer takes advantage of the significant open source technologies that dominate this field. With thousands of developers focused on these projects, the result is inexpensive, robust and solid cloud platforms.

Backend performance is imperative, but for front-end performance, usability is a mandate. Cognizer is designed to allow users to perform their work easily, on many devices and in many formats. Design is critical not only for look and feel, but for information capture, flow and validation.

### Enterprise Architecture

Once the cloud services began to reach critical mass and adoption, the design patterns for programming began to change to reflect this new environment. All of Cognizer's functionality is implemented as services that are virtualized and implemented as containers. They can be deployed independently to provide elastic scalability, reliability and fault tolerance.

The Cognizer system was implemented as a set of micro services. These are stateless, small, message-enabled, bounded by context, autonomously developed and independently deployable. They are also built and released with automated processes. Each micro service is a self-contained piece of business logic with a defined interface, specific functionality and clear business value.

### Enterprise Deployment

Finally, IT management demands that modern software be installable, maintainable and manageable. Cognizer uses advanced solutions like Docker, Ansible and Kubernetes. This allows IT to use automation to deploy, configure and manage. In addition, they require advanced logging, monitoring and access to events in real-time.

The bar is high for IT because the bar is high for users. Perfect execution is expected, and anything below perfection is considered a failure.

### Cognizer Applications

In the Corporate Brain platform, Cognizer has developed three very useful applications that are not only simple to use, but also provide the user with tremendous value.



### Brain Assistant

Brain Assistant is a powerful Conversation AI for the Enterprise. You can think of it as “Jarvis” for your company. It knows virtually everything, both inside and outside the company. You can ask it about information in your email, document repositories, collaboration systems, and Enterprise applications like Salesforce and Zendesk.

When the Brain Assistant answers the question, it not only gives you the answer, but also provides the section of text where the fact was found, with the answer highlighted. If you click on the section of text, it launches the underlying application and brings you right to the specific page of the information.

It also allows the user to give feedback to the Brain in the form of thumbs up/down and a comment. This information is then used to optimize the answer in the future and provides feedback in the Cybrarian Console (see Cognizer Cybrarian — Our Human in the Loop Solution White Paper).

### Brain Waves

Brain Waves is designed to listen to your conversations, on the phone, in a meeting or just casual discussions. As you talk, Brain Waves is looking for important words. When one is spoken, it puts a tile on the screen with a few facts about that word. If you need more information, select the tile and a full Dossier is provided about the word.

This is very valuable in a conversation that mentions important people, new technologies, acronyms or anything you might not know. Instead of asking about the information or doing a Google Search, Brain Waves provides the information in context and in real time. This makes you much smarter in the conversation and makes the conversation more valuable.

If desired, the conversation can be recorded and transcribed. This information is then digested by Cognizer and stored in the HyperGraph.

### Brain for Box.com

Brain for Box.com provides deep integration with the Box Enterprise Content Management System. It is implemented as a Chrome Extension and occupies about 20 percent of the right side of the user experience. When you use the search box or ask an important question, Box conducts the search and displays the results in the left side of the UX. But in the Answer Box on the right, the actual answer to the question is displayed, along with the section of text where the answer was found. Instead of looking for the



answer in what could easily be 100 documents in the search result, Answer Box does all the work for you.

### Custom Applications

As valuable as the Cognizer applications are, the Brain platform can provide even more value when developing custom Use Case Apps. There are many forms of these applications and each can be powered by the Brain platform.

### Q&A Applications

Q&A Applications are the most popular of these custom use cases. This might power a Bot on the company's Customer Support Website, provide an Employee Concierge or provide an Expert System to a business-critical system.

No real programming is required for most Q&A Systems. Just upload the information that will be requested, and Cognizer will be up and running. If there are any specific requirements, the system can be tuned by using the Glossary and Cybrarian Console.

### NLP Applications

NLP Applications can be developed using Cognizer's REST/JSON interface and its Micro Services. Cognizer can develop these applications, or they can be developed by service providers or even your Company's development team. The AI Models are completely abstracted, so it does not require a data scientist, just application development and the ability to communicate with the platform.

These NLP Applications can be very powerful. Cognizer can answer questions, search for information by keyword, complete calculations and even navigate very complex tables. Cognizer is not a basic NLP Service. It is a full platform with hundreds of NLP modules, fully integrated with enterprise security, management, and monitoring. Cognizer basically turns all of the content in your organization into a quadrable database, and you can build applications against it like any other database.

### Aggregation Applications

Cognizer can even aggregate information within a document or across thousands of documents. One example: If a property management company had hundreds or thousands of leases, you could ask Cognizer for the total amount of space in all of the buildings, the total space currently leased or the total space available to lease.

Very sophisticated math can be applied because Cognizer is tightly integrated with Wolfram Alpha's Mathematica Engine. The system can calculate advanced math, calculus, linear algebra, and even advanced business and engineering formulas.



### Time Series Applications

Finally, Cognizer can provide powerful Times Series analysis. Since all of Cognizer's facts are stored in a triple-store database, time series dimensions can be materialized to show the changes of information over time.

## Conclusion

Large enterprise IT requirements have come a long way since Web technologies began to be massively adopted in the early 2000s. We now think of state-of-the-art as containerized, services and microservices, cloud deployed, open source frameworks, and web scale.

Cognizer has very bold ambitions, and we feel we are very well-positioned to take advantage of the next wave of enterprise technology. In this next evolution, our information will become smart and interactive, providing more value than we have ever seen before.

We believe:

- The Corporate Brain (The Brain) will be the next evolution of enterprise optimization.
- The Brain will be implemented as a platform and will be available as a service to applications throughout the enterprise.
- The Brain will not only connect to our current enterprise applications, but to our enterprise content stored in documents, emails and collaboration systems.
- The Brain will leverage both Open Source and Optimizing Models to provide state-of-the-art results and performance.
- The Brain will be based on the latest web and enterprise technologies, including microservices, deployed as containers, cloud-enabled, and managed with advance systems like Docker, Ansible, and Kubernetes.
- The Brain will need to be Web scale, as it will not only support internal resources, but also our customers and partners.

We invented this category, and there is only one solution in the world that meets these requirements. Cognizer, the Corporate Brain, is the solution to this problem, and we are engaging with innovative customers and partners to introduce the Intelligent Enterprise!