UDC 004:665.6/.7
Preliminary Communication

Applying Genetic Algorithms and RIA Technologies to the Development of Complex-VRP Tools in Real-World Distribution of Petroleum Products

Antonio Moratilla Ocaña *antonio.moratilla@uah.es*
Computer Science Department, Alcalá University
Alcalá de Henares, Spain

Eugenio Fernández Vicente *eugenioj.fernandez@uah.es*
Computer Science Department, Alcalá University
Alcalá de Henares, Spain

José Ignacio Olmeda Martos *josei.olmeda@uah.es*
Computer Science Department, Alcalá University
Alcalá de Henares, Spain

Juan José Sánchez Peña *juanjo.sanchez@uah.es*
Computer Science Department, Alcalá University
Alcalá de Henares, Spain

Alberto Moratilla Ocaña *alberto.moratilla@uah.es*
Computer Science Department, Alcalá University
Alcalá de Henares, Spain

Abstract

Distribution problems had held a large body of research and development covering the VRP problem and its multiple characteristics, but few investigations examine it as an Information System, and far fewer as how it should be addressed from a development and implementation point of view. This paper describes the characteristics of a real information system for fuel distribution problems at country scale, joining the VRP research and development work using Genetic Algorithms, with the design of a Web based Information System. In this paper a view of traditional workflow in this area is shown, with a new approximation in which is based proposed system. Taking into account all constraint in the field, the authors have developed a VRP Web-based solution using Genetic Algorithms with multiple web frameworks for each architecture layer, focusing on functionality and usability, in order to minimizing human error and maximizing productivity. To achieve these goals, authors have use SmartGWT as a powerful Web based RIA SPA framework with java integration, and multiple server frameworks and OSS based solutions, applied to development of a very complex VRP system for a logistics operator of petroleum products.
Keywords: RIA, SmartGWT, Web Framework, Web Architecture, Genetic Algorithms, VRP.

1. Introduction

Transport of goods and people is the biggest problem domain in logistics, and so it represents a substantial task in the activities of many companies. The main problem in this field consists in defining the optimal routes for a fleet of vehicles to serve a set of clients, and is known as the Vehicle Routing Problem (VRP), which consists in delivering goods to a set of customers with known demands through minimum-cost vehicle routes originating and terminating at the depot. The utilization of computational systems often results in significant savings in the total costs. There are several variants of the VRP [1, 2], [4], [5], [7], [12], [15, 16], [19]. In order to solve the problem, an extensive group of techniques have been addressed, and they can be classified into three categories: exact, heuristic and metaheuristic methods. All these techniques increase computer-time in order to obtain required optimization results and it is necessary to develop sophisticated programs in order to reduce the computation time.

In this paper we present a VRP web-based system using Genetic Algorithms. The system allows planning engineers to generate a near optimal vehicle assignment and routing plan based on daily shipping demands. The improvements compared to the current situation, embrace the maximum occupancy of trucks, total travelled distance, reduction of delivery times, minimization of distance travelled with minimal or no occupation of the truck, etc.

In terms of computer-time, the contribution of this work is to define a Genetic Algorithm that is able to compete with the best known approaches in the resolution of the VRP in terms of the solution found, the execution time, and the number of evaluations made. In addition, it represents a kind of paradigm much simpler of customization and comprehension than others such as Tabu Search or very specialized hybrid algorithms.

From the standpoint of information systems, over the years, a large body of research and development covers the VRP problem and its multiple characteristics, but few investigations examine it as an Information System, and far fewer as how it should be addressed from a development and implementation point of view.

This paper tries to address this situation by describing the implementation decisions made by the authors toward the development of an Information System for a VRP solution. At the conceptualization phase of the system, authors have been aware of some needs that must be fulfilled to get the system on duty: every order must be served, not matter how hard, long or difficult it is; security on handling and transportation is key; traceability of operations is a must; environmental concern is inside all business process; multiple areas of distribution, with little to none common frontier; constant changes of customer orders.

Web technology evolution has been a constant from the beginning of the field: changing from CGI (Common Gateway Interface) applications to SOA (Service Oriented Architecture), from simple HTML to RIA (Rich Internet Application) [11] SPA (Single Page Application) [22] and from full page responses to REST services. This evolution has been going along with a strong and growing OSS (Open Source Software) community, which has driven great projects, from infrastructure project as Apache HTTP Server, Apache Tomcat, Jersey, Spring, etc., to high level projects as OSM (Open Street Maps, [27]), that competes with the biggest corporations and offers a great service.

From development point of view, the creation of frameworks like GWT (Google Web Toolkit), SmartGWT [33], GXT, ExtJS, MooTools, EmberJS, AngujarJS, etc., are a huge advance in user interface programming tools and capabilities. These frameworks drive the innovation at UI level from simple HTML pages to responsive and dynamic RIA SPA (like the ones used at Twitter, Google plus, Gmail). This kind of RIA SPA leads the user to behave as with native applications, blurring the lines between web and local applications on perception level [28]. From these frameworks, SmartGWT and ExtJS are the best suited for RIA SPA enterprise class web applications, as they have extremely powerful widgets that drives user interaction in a native application way. SmartGWT is especially useful as it can be integrated into a Java web project, and compiled as a whole, making easy to develop and debug its applications. Other projects like Apache Struts 2, Jersey or Spring in Java language, or Epiphany and Symfony2 in PHP, are examples of enterprise accepted frameworks developed under OSS environment. These frameworks are designed to make easy web development of big scale architecture solutions, using standards as JSON for data interchange or software architectures styles like REST for service layer design. The use of JSON is getting higher these days because of its simplicity and low memory footprint, which makes it ideal to be used on mobile devices with constrained resources.

At storage level there have been huge pushes on lasts years, mainly because of new NoSQL databases, and referred to web development it makes sense to look at document oriented storage solutions like MongoDB [23] or CouchDB [8]. Trying to replicate the way those works, there are traditional relational database solutions like PostgreSQL [30] adopting

JSON as a native data type in their schemas, even allowing querying JSON data from SQL queries. As persistence solution, document oriented NoSQL databases has been integrated into solutions like Spring Data, and using Spring Data with Hibernate, while others solutions like MyBatis [24] provides a lower level of abstraction and do not try to integrate NoSQL Solutions. Moreover, OSS project at higher level, as OSM, with a huge growing community providing data and support, has been able to build services with great added value, driving innovation with newer and focused projects like Leaflet JS [20], Nominatim [25] or OSRM [21]. These projects are based on OSM file format definition [29] and by taking data from that format, they're the edge of innovation on web mapping applications, allowing building a service to route people from point A to B in an easy, efficient and fast way.

Actually, the use of these technologies enables us to implement highly complex applications with low development time, such as advanced route optimization, delivery scheduling and logistics planning software. During the last decade, there has been acknowledged a tremendous change in enterprise-oriented business software where traditional products have gradually left their place to integrated, friendly, usable and efficient solutions that would rigorously deal with every single business aspect of each individual enterprise. The VRP system presented in this article belongs to this category, as it is developed on SmartGWT, MyBatis and Spring Web-Development platform and possesses modular and flexible structure. Web-based techniques are less expensive, more efficient and lately have been the target of most development efforts. On the other hand, web-based solutions are able to easily interoperate with the whole supply chain entity. The development of this system has been possible thanks to the use of SmartGWT, Spring, MyBatis, OSRM, Leaflet and OpenStreetMap technologies.

2. CHRONOS

In this paper we approach the VRP problem of one of the biggest Spanish companies for oil distribution products by road. The enterprise is located in different areas of Spanish territory and operates on great public and at the same time has several individual customers. Every day, several times a day, a customer network needs to be serviced by a fleet of heterogeneous capacitated vehicles located on a several depots or distribution centers, subject to the high number of constraints, such as:

- The vehicles are multi-depot or single-depot.
- The capacity of a vehicle cannot be exceeded.
- A single vehicle supplies each customers demand.
- The number of vehicles used is pre-determined.
- Schedule of the drivers must be respected.
- The number of drivers is pre-determined; the time to serve the customers should be respected.
- The shipping demand of a depot cannot be divided. It should be delivered by the same vehicle, unless the shipping demand of the depot exceeds the loading capacity of a vehicle.
- The time-window constraint is known, the driver will be given the time limitation so that over-time driving can be eliminated.
- The cargo loading cannot exceed the vehicle loading capacity at each delivery.
- There are over ten different types of products, and some of them are incompatible in the same lorry.
- Etc.

Given the area of a country like Spain, it's not logical to set a unique operation center for fuel distribution. There are some factories/refineries settled along the country in key places where the fuel products are ready to serve to the final distributor (gas stations, large buildings, etc.). The fuel distributor used to set a base for their trucks near the factories/refineries, so it's faster and cheaper to make fuel distribution this way. With this configuration in mind, the business is ruled by resources localities, and the planning of an area is held by plan engineers sited in that area. This gives better understanding of area based constraints and risks, and makes easier to plan accordingly to the most up to date fleet and customer information. All distribution orders are clustered in a time-zone way, and must fulfil a planning and confirmation process before they could be served.

A Complex-VRP system (CHRONOS) using Genetic Algorithms and RIA development technologies is designed in order to automatically generate vehicle routes, which vehicles should deliver to which customers and in which order, minimizing simultaneously the vehicle cost and the total distance travelled by the vehicles. Over the years, a large body of research and development covers the VRP problem and its multiple characteristics, but few investigations examine it as an Information System, and far fewer as how it should be addressed from a development and implementation point of view. Here, we tries to address this situation by describing the implementation decisions made by the authors toward the development of an information system for a VRP solution (CHRONOS), applied to one of the biggest Spanish companies for oil distribution products by road.

2.1. Solving VRP with Genetic Algorithms

The distribution problem in general started with two classical problems in combinatorial optimization: the Traveling Salesman Problem (TSP) and the Vehicle Routing Problem (VRP). The TSP consists, from a departure point, to visit a set of customers with one single truck and to come back planning its tour by finding the sequence of customers with the lowest possible total cost. Historically, [10] is the first work that introduces TSP problem by proposing resolution's methods. The vehicle routing problem addresses the case where each customer has a given request. It consists in determining several tours that all start and end at the depot and where each customer is visited once by a single truck. The first work that addresses the VRP is [9]. The VRP generalizes the traveling salesman problem (TSP) and is much more difficult to solve than the TSP [17]. As we have seen, the physical distribution problem in general, is not a recent problem. As a component of the supply chain, includes a set of activities executed to obtain the delivery of a product from the production location to the end customer. Problems related to physical distribution are: selection of distribution channels, determination of customer service level, distribution centers, location planning, inventory management, transportation means selection, fleet composition, delivery scheduling and vehicle routing, etc., and the objective is double: to minimize the total transportation cost while rationalizing the vehicles utilization. In this sense, vehicle routing refers to a broad group of problems that could be expressed as following: a finite set of customers at fixed locations with defined demand, must be supplied with goods by a number of vehicles having a finite capacity and predefined starting points and terminals.

The classical VRP is defined as follows: Let G = (V, A) be a directed graph where V = $\{0, \cdots, n\}$ is the vertex set and $A = \{(i, j) : i, j \in V, i \neq j\}$ is the arc set. Vertex 0 represents the depot whereas the remaining vertices correspond to customers. A fleet of m identical vehicles of capacity Q is based at the depot. The fleet size is given a priori or is a decision variable. Each customer i has a non-negative demand q_i. The principal decision variables are:

- T_i arrival time at node i
- w_i wait time at node i
- $x_{ijk} \in \{0,1\}$, *0 if there is no arc from node i to node j, and 1 otherwise,* $i \neq j; i,j \in \{0,1,2,\ldots,N\}$

The parameters are:

- *K total number of vehicles*
- *N total number of customers*

- *c_{ij} cost incurred on arc from node i to j*
- *t_{ij} travel time between node i and j*
- *m_i demand at node i*
- *q_k capacity of vehicle k*
- *e_i earliest arrival time at node i*
- *l_i latest arrival time at node i*
- *f_i service time at node i*
- *r_k maximum route time allowed for vehicle k*

The objective function minimizes the total cost of travel of all the vehicles in completing their tours.

$$\sum_{I=0}^{N}\sum_{j=0}^{N}\sum_{j\neq i,k=1}^{K} c_{ij}\, x_{ijk}$$

The vehicle routing problem literature is abundant. At the origin of routing problems, [18] provided a bibliography of 500 studies. In [13] is presented a VRP literature classification, based on a review of about 1,500 documents. Extending the basic VRP approach, numerous variants have emerged over the years, among which, the most discussed are:

- CVRP ("Capacitated VRP"): *each vehicle has a limited capacity* [31].
- MDVRP ("Multi-Depot VRP"): *the seller uses several depots to supply customers* [15].
- PVRP ("Periodic VRP"): *orders can be taken only on certain days* [4].
- SDVRP ("Split Delivery VRP"): *customers can be served by different vehicles* [1], [12].
- SVRP ("Stochastic VRP"): *some values such as the number of customers, their demands, length of service or travel time are random* [19].
- VRPB ("VRP with Backhauls"): *customers can return the goods* [16], [31].
- VRPPD ("VRP with Pick-Up and Delivering"): *customers have the option to return some goods to depot* [32].
- VRPSF ("VRP with Satellite Facilities"): *vehicles can be supplied without returning to the central depot for other auxiliary route* [5].
- VRPTW ("VRP with Time Windows"): *each customer has to be served within a certain time window* [7].

The VRP is so widely studied because of its wide applicability and its importance in determining efficient strategies for reducing operational costs in distribution networks. Today, exact VRP methods have a size limit of 50 - 100 orders depending on the VRP variant and the time-response requirements. VRP problems are within the combinatorial optimization problems, and from the standpoint of computational complexity, is one of the most complex because it is NP-Complete kind of problem: It cannot be solved in polynomial time [3], [14], [36].

In order to solve the VPR problem, a group of techniques have been addressed. They can be classified into three categories: exact, heuristic and metaheuristic methods. We can say that the exact methods are efficient in problems up to 50 depots [2] due to the computational time constraints. Furthermore, heuristic methods provide us with acceptable solutions obtained by a limited exploration of the search space. A review of these can be found in [26]. Finally, the metaheuristic techniques, developed in the late 90s, are characterized by performing a search procedure to find acceptable solutions by applying domain independent operators that modify intermediate solutions guided by the suitability of its objective function. Within these Neural Networks, Tabu Search, Genetic Algorithms or Ant Algorithms can be found, among others. A review of these methods can be seen in [6].

In order to select the correct heuristic we consider four criteria: accuracy, speed, simplicity and flexibility. Accuracy measures the degree of departure of a heuristic solution value from the optimal value. With respect to speed, real-time applications require fast, sometimes almost instantaneous, action. On the other hand, several VRP heuristics are rarely implemented because they are just too complicated to understand and to code. Finally, a good VRP heuristic should be flexible enough to accommodate the various side constraints encountered in a majority of real-life applications. In this paper, Genetic Algorithms are used to solve the VRP proposed, because we consider that it meets the above four conditions (accuracy, speed, simplicity and flexibility).

Genetic Algorithms have been inspired by the natural selection mechanism introduced by Darwin. They apply certain operators to a population of solutions of the problem at hand, in such a way that the new population is improved compared with the previous one according to a specified criterion function. This procedure is applied for a preselected number of iterations and the output of the algorithm is the best solution found in the last population or, in some cases, the best solution found during the evolution of the algorithm. In general, the solutions of the problem at hand are coded and the operators are applied to the coded versions of the solutions. The way the solutions are coded plays an important role in the performance of a genetic algorithm. Inappropriate coding may lead to poor performance. The operators used by genetic algorithms simulate the way natural selection is carried out. The most well-known operators used are the reproduction, crossover, and mutation operators applied in that order to the current population. The reproduction operator ensure that, in probability, the better a solution in the current population is, the more (less) replicates it has in the next population. The crossover operator, which is applied to the temporary population produced after the application of the reproduction operator, selects pairs of solutions randomly, splits them at a random position, and exchanges their second parts. Finally, the mutation operator, which is applied after the application of the reproduction and crossover operators, selects randomly an element of a solution and alters it with some probability.

The basic steps of genetic algorithm are:

- Encoding: we represent each city with number. For N number of cities, the cities are represented by permutation of integers from 1 to N.
- Initialization: an initial population is generated from many individual solutions. The search space contains all the possible solutions from which the population is generated randomly. However, the solutions are seeded in the areas from where the optimal solutions are likely to be found. In VRP problems, if there are N numbers of cities then N! possible number of solution can be made.
- Fitness Function: in this each solution or chromosome is assigned a fitness value. For VRP problem, the fitness value is assigned to each solution by calculating the distance between the cities in each solution.
- Selection Operator: simulates the natural law of the survival of the fittest in the population evolutionary process. The chromosomes selected with largest fitness value. The solution with minimum distance is selected for crossover operator. Here is possible to use different selection techniques.
- Crossover Operator: reproduction makes clones of good strings but does not create new ones. Here is possible to use different crossover techniques.
- Mutation Operator: after the crossover operation, the chromosomes are subjected to mutation process. Mutation prevents the algorithm to be trapped in local optima. Here is possible to use different mutation techniques.
- Termination: when the algorithm has run a given number of iterations, it stops and output the best solution. This generational process is repeated until a termination condition has been reached.

Tens of selection, crossover and mutation operations can be used. The selection methods adopted in this paper are: Roulette Wheel Selection (RWS), Tournament Selection (TS) and

Stochastic Universal Sampling (SUS). The crossover operators are: Partially Matched Crossover (PMX), Cycle Crossover (CX) and Order Crossover Operator (OX). Finally, the mutation operators are: Displacement Mutation (DM), Exchange Mutation (EM), and Simple Inversion Mutation (SIM). We have used Matlab and C++ programming language to implement out system. The main advantages of Matlab is that provides immediate access to thousands of fundamental and specialized genetic algorithms functions written by experts, in addition to those written by yourself or your colleagues, and is possible to convert Matlab to C++ code.

In order to obtain the best operators selection, we checked our engine with different public datafiles (A-n32-k5.vrp; att-n48-k4.vrp; E-n76-k7.vrp; B-n43-k6.vrp in http://neo.lcc.uma.es/vrp/vrp-instances/) and execute several independent runs with different combinations of operators. For a particular problem and a particular algorithm, a run is said to be successful run here if the best objective function value found in that run lies within 5% accuracy of the best known objective function value of that problem. Some solutions were quite good compared to published values on the internet, while others were not really satisfying (Figures 1 and 2).

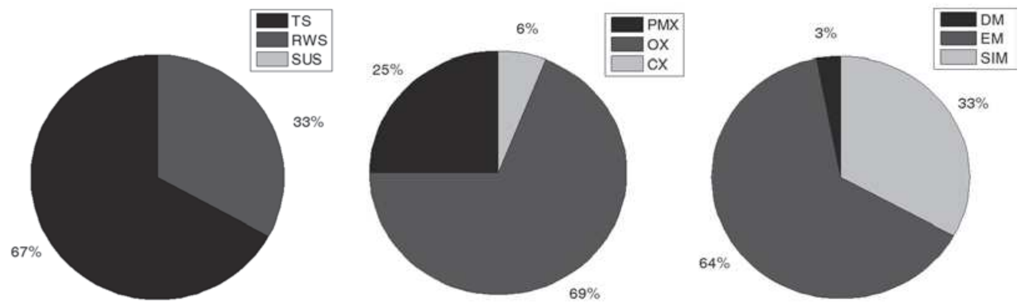

Figure 1: Results in terms of distance

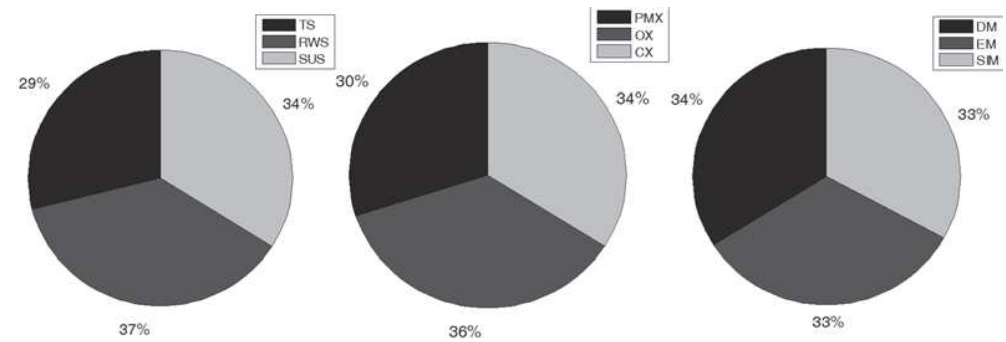

Figure 2: Results in terms of speed

Combining results in terms of distance and speed, we conclude as result, that Tournament Selection (TS) with Matched Crossover (PMX) and Exchange Mutation (EM) are the best combination. Using Matlab and C++ the core of CHRONOS in terms of optimization is built using Genetic Algorithms with TS+PMX+EM operators.

2.2. Using RIA Technologies to Develop CHRONOS

Before CHRONOS, the process to complete a day by day distribution was conducted by a myriad of programs, leading to high inefficiencies and even duplicated process, error prone. The situation gets even worse if human resources management gets into consideration: all processes implemented in the workflow are isolated from a planning engineer to another because of tools used cannot share the acquired knowledge. This process, as shown in Figure 3, implies multiple external systems and programs.

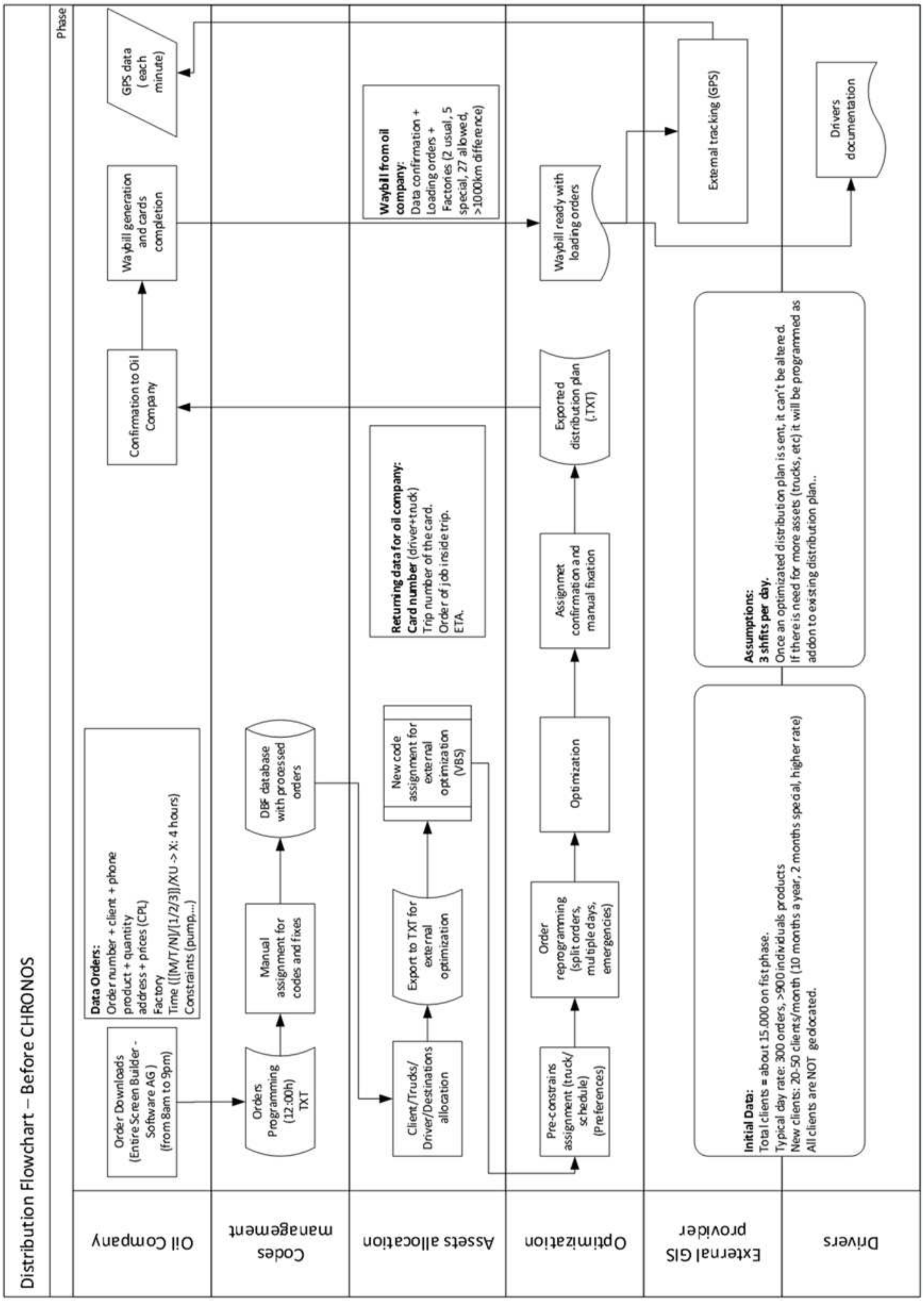

Figure 3: Initial workflow, isolated for each planning engineer.

Given the initial constraints of the problem, CHRONOS proposes a new unified workflow that combines the knowledge of all planning engineers and their experience into a unique shared web system (Figure 4):

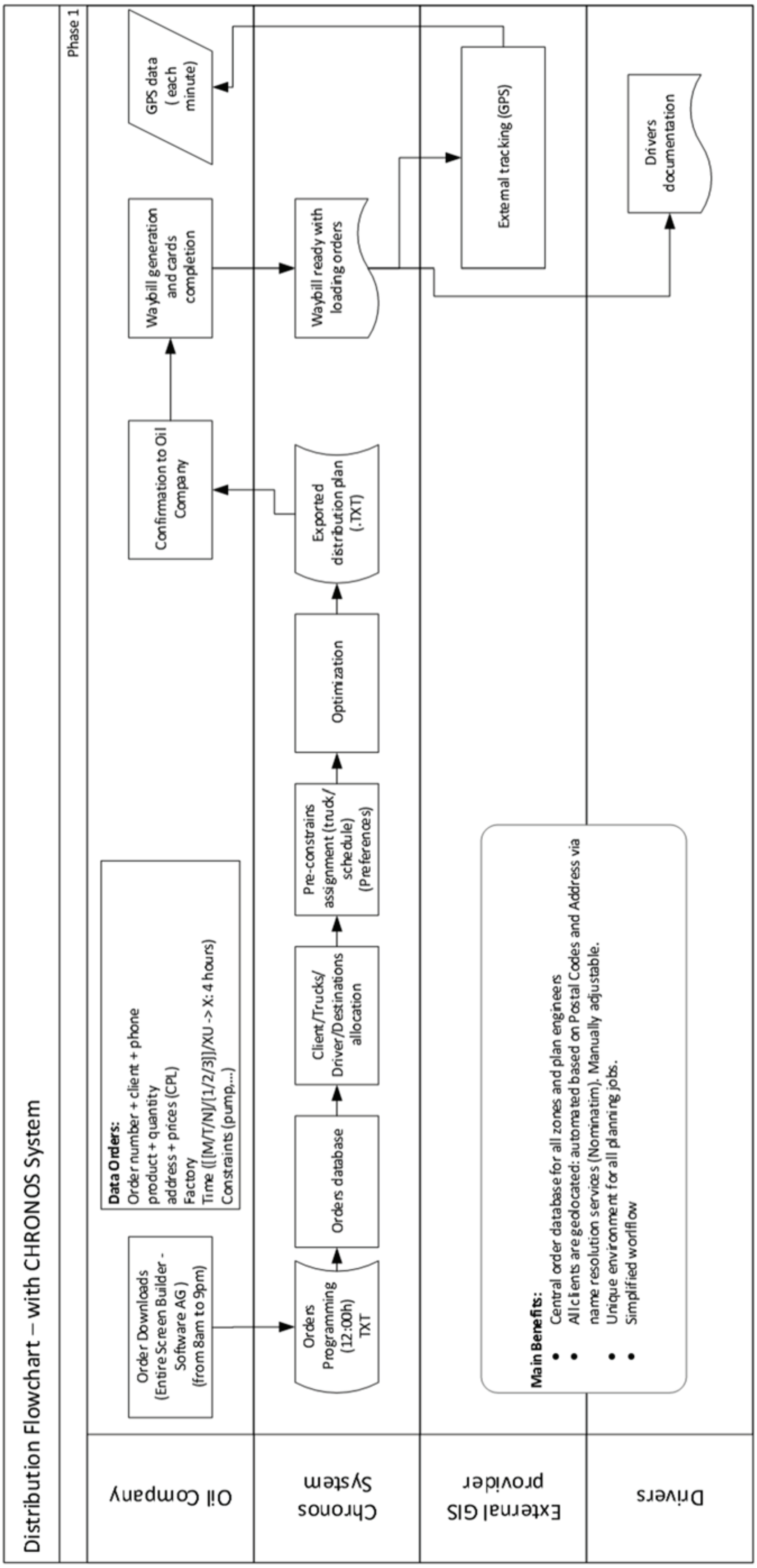

Figure 4: CHRONOS unified workflow

With the chosen approach (Figure 4), CHRONOS was able to reduce the effort of engineers to 50% of the original effort, reducing from 2 hours to 1 hour the time to do a distribution plan, while the error probability is minimized due to shared knowledge.

Even more important is that improvement in the time to do planning is greater as the system is used, as the adjustments and restrictions of the various system assets (trucks, drivers, locations, etc.) are stored and shared, ready to be used by the next planning engineer without having to re-enter them or keep them updated multiple times as was the case previously. In this case, it’s been found that time to complete a distribution plan goes down to 20-25% of the original effort.

This has an added benefit: any planning engineer can make planning for any area with minimal contextual information, allowing solving unexpected problems, failures or peak workloads seamlessly.

CHRONOS is composed of multiple modules working as a whole to achieve the distribution’s plan. A distribution plan is shown in Figure 6, where the map of a city and all destinations and routes, using the Leaflet framework, is shown. Taking into account all modules (Figure 5), the Chronos System has a management module, an ERP module, an Orders module and an Optimization Module. These modules are integrated on the System as first class citizen, as its functionality is needed to make the system work as it should. There are other modules like HCAE or RTCM (Real time Control Module), that are not indispensable to the planning work, but are needed for subsequence business process.

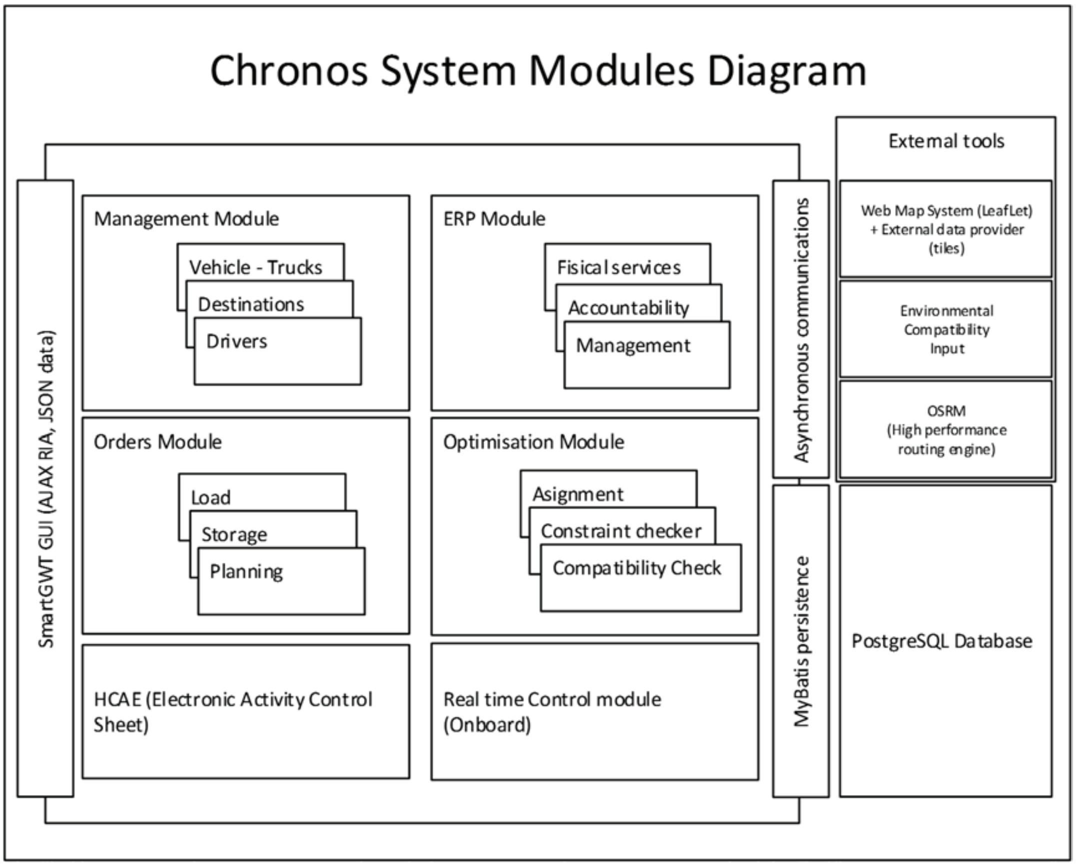

Figure 5: CHRONOS Modules diagram.

Other external modules are the webmap system, generated with leaflet framework and integrated into SmartGWT and Spring, and the OSRM module as high performance routing system installed on a different machine. All modules are connected to the central database, where all information is kept in a always coherent way, warrantying all data is up to date and

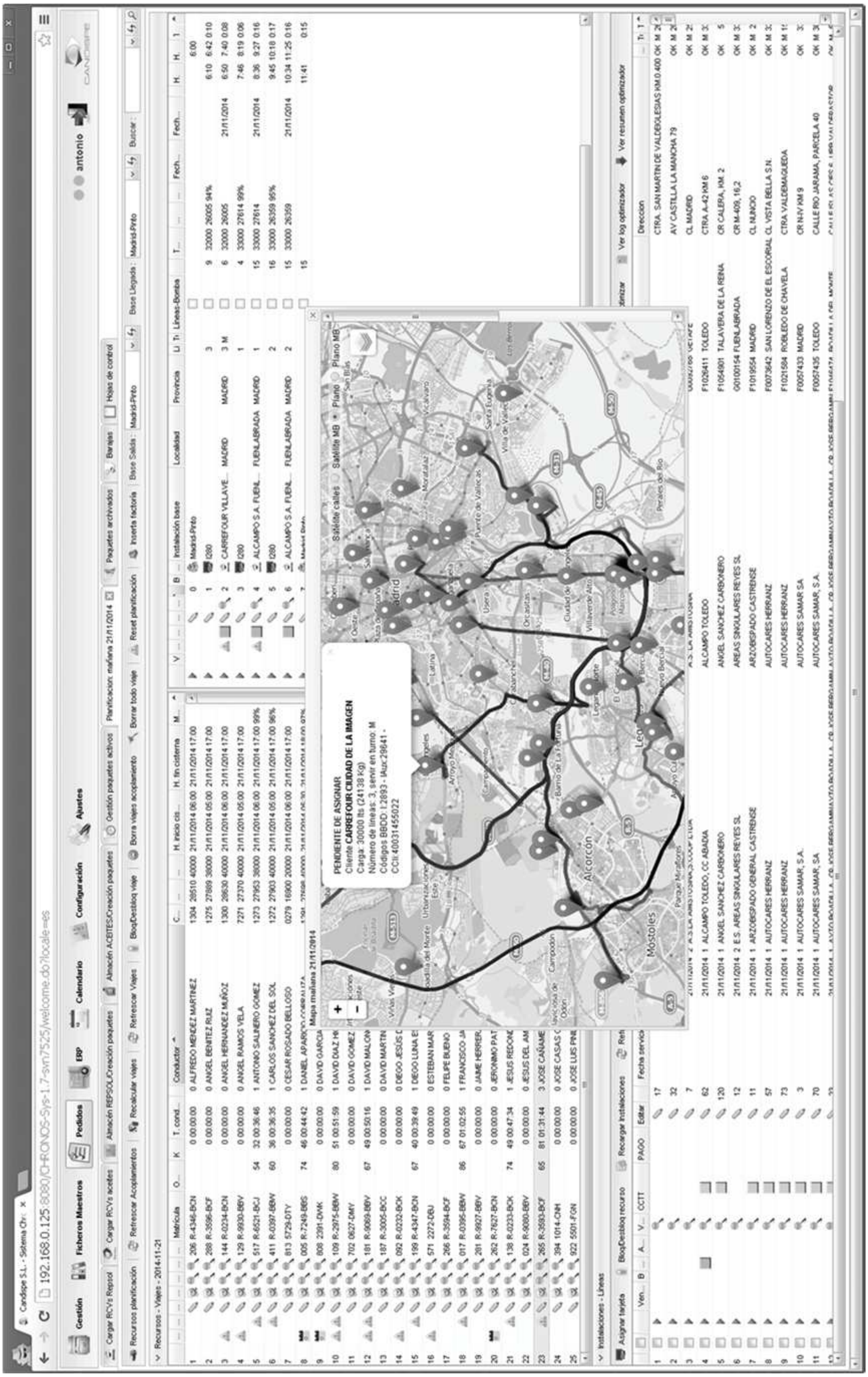

Figure 6: Example: planning of a fuel distribution.

ready to use in VRP optimization process. To develop all these modules, a combination of SmartGWT, Spring, MyBatis, LeafletJS, OSRM, OSM and PostgreSQL are used.

At the conceptualization phase of the system, authors have been aware of some needs that must be fulfilled to get the system on duty:

- Every order must be served, not matter how hard, long or difficult it is.
- Security on handling and transportation is key.
- Traceability of operations is a must.
- Environmental concern is inside all business process.
- Multiple areas of distribution, with little to none common frontier.
- Constant changes of customer orders.

By identifying the needs exposed before, the system arises some of its key areas, where we will be proving solutions:

- Multiple planning locations, usually based on truck depots.
- Simple routing from point A to B.
- Complex routing from multiple points with constraints.
- Manual operation.
- Integrated information of all resources.
- Non-blocking operation.

On the other hand, CHRONOS must take the locality features into account and be able to operate in multiple locations, while sharing all data about fleet and customer so managing staff can get over any problem could arise in day by day operation.

These constraints lead to development of a Web-Based Information System, so it can serve to multiple clients on multiple locations. By contrast, the approach used to the date on Spanish fuel distributor is the use of single seat software, with local databases, unshared or hard to share information, and local optimisation software.

The objective of simple routing is to provide a route from point A to point B with 3 information components: distance, time and intermediate GPS coordinates at a fixed zoom level. While it seems a simple problem, there are some initial constraints which must be addressed because simple routing is used for two different objectives:

- GUI guidance to planning engineers in order to visualize the real route its trucks will cover on a map.
- Create a distance/time matrix, so it can be used on VRP optimization.

Simple routing calculation is handled by OSRM component. This component is built by a dedicated web server on a dedicated machine of the information system. It loads all map data into RAM memory, so it won't use the disk when OSRM is on duty. It can handle multiples route queries, but it's limited on the amount of CPU cores and how the requests are made.

We have measured 900 request per second with a Intel Core i7 3770 CPU, and while it's a good result, it's not good enough to keep the pace with a VRP problem, NP-hard by nature. The distance matrix creation problem gets harder as new destinations are added to the problem, so a VRP problem with 1 depot and 39 destinations gets n2-n viable routes: 1.560. But if the number of destinations grows 199, the number of viable routes goes up to 39.800 routes. We have found the number of destinations range from 200 to 500 in real world problem, and it must fulfil orders for about 5 million litres per planning job and zone. These lead us to a worst case scenario of 249.500 viable routes, and that could suppose more than 4 minutes only to get ready to launch VRP algorithm process.

To address this situation a route cache was built. All route queries made by CHRONOS go through a database backed cache system to reduce the number of queries made to OSRM component. This approximation gets three key benefits: first, the distance matrix creation process is much faster as only new destinations must be processed. Second, the routes could be tweaked offline to adapt them to business needs (tolls, dangerous routes, legal limitations, etc.). Third, OSRM component gets more available time to be used by other users on their planning jobs.

At the time of writing this paper, the cache system is running with more than 17 million routes, which are refreshed periodically to reflect map's changes and optimizations.

As long as CHRONOS is designed to fulfil the needs of fuel distribution logistic, it must check some compatibility issues when serving the orders to the clients. These issues go from checking capacity of the truck, to marketing compatibility or even contamination episodes with environmental consequences.

Nevertheless, all those checking must be pledge to plan engineers will, because there are cases that get out of control easily and must be addressed. These cases go from a human error while handling the fuel, to an emergency call for fuel to firefighter's helicopters or planes, or to terrorist attack to factories and/or depot facilities.

Given that these situations are completely out of usual behaviour, cannot be forecasted, and the way to resolve them depends on the situation itself, CHRONOS must be able to obviate all its systems checks and usual capabilities so plan engineers can do their distribution plans as they need with the maximum help and minimum limitations from CHRONOS.

While almost all Information Systems tend to integrate all business information, this goal is key in a VRP environment. Within a VRP problem there are multiple constraints which must be testes, checked and evaluated to be able to know if a result is a valid/viable result or if it should be discarded. The present case is extremely difficult in these areas, as it requires multi-element cross compatibility checking, as it may need to check if the products can be downloaded by a pump pulse, or how many liters are wasted on the discharge hose if has a longitude or another and if that quantity will lead to chemical incompatibility with the next download. At the same time, all business distribution resources must be compatible checked with customers (e.g. it's impossible to serve an order with a 3 axis truck in a historical town center, it's too big. There're other constraints with hose length, or with some military facilities). The system must observe and preserve the legal constraints on drivers rest, and must find who is the best driver for each order, given his work calendar and other constraints (holidays, inactivity due to medical reasons …)

As shown before, all information is needed to be available to VRP so it can pick a good solution to optimization problem, but take care of all information implies that storage information system must fulfil some needs:

- It must be ACID compliant: multiple changes must be taken care as one atomic change, even if they affect multiples objects/entities: this influences the choices available, as Document Oriented NoSQL Databases do not support ACID at database level at this time(only at document level, which it is not enough).
- The system should be able to be used easily in web based solutions: from persistence view, it should integrate with a persistence solution, and from data logic view inside web service layer, there will be data which should not be processed, but passed, so there's a clear advantage to store it in its original format.

While Web UI has gone a long way to lead to non-blocking interaction, using Ajax to be able of sending and receiving multiples request to a web server, and it's now on common use, non-blocking operation inside the service layer of a web server is not so common, and leads to unusual situations.

At a VRP Information System there are some processes that require some time to complete. Given the need of information to create the data needed to a VRP optimization, it's easy to understand that it may take time to complete.

In order to address these situations, CHRONOS works with asynchronous actions and a pool of semaphores to route the internal logic at service level. By taking this approach, when CHRONOS receives a request on one of its asynchronous actions, it split to logical routes inside server code, one to respond with the state of the asynchronous job requested, and other to complete the asynchronous job if it was not already at work.

Hence, the system has a pool of asynchronous jobs executing that can be managed by users. Those jobs go from creating a distance/time matrix, to control the VRP algorithm as an external process, or to update at night the whole route cache system if it's needed.

Through the use of the developed VRP system, the enterprise obtained the following results:

- Significant cost-reductions and time-savings.
- Improvement of the maximum occupancy of trucks.
- Reduction of the total travelled distance.
- Reduction of delivery times.
- Minimization of distance travelled with minimal or no occupation of the truck.
- Improving business processes of major importance.
- Reducing personnel's occupation times.
- More flexible and efficient planning.
- Improved communication and data, transfer of critical information for the whole enterprise.
- Instant access to real-time data.
- Etc.

3. Conclusions

In this paper we introduce the development of a web system, named CHRONOS, witch aim to simplify and optimize the day by day work of a fuel distribution company, while addressing new needs on environmental and capacity constraints. To develop this system, we've used Genetic Algorithms to solve VRP problem, SmartGWT framework for RIA SPA UI, Spring for service level architecture at web server, MyBatis as persistence solution, PostgreSQL as database backend, OpenStreetMaps as map data source, OSRM as high performance routing engine and LeafletJS as map visualization framework, building a technological stack ideal for VRP systems. The use of these technologies has allowed the development of a web-based complex vehicle routing planning system that helps the company to solve the daily vehicle routing problems. The system routing problem developer for the case company was formulated based on the company's current delivery network, and is able to reduce the operating costs and increase the competitiveness of the company. With the aid of this system, the time needed to generate a routing plan is significantly reduced in comparison with the time needed in the older planning, and helps to shorten the learning curve for new staff in dealing with the routing process. As a result Through the use of the developed VRP system, the enterprise obtained as results: significant cost-reductions; improvement of the maximum occupancy of trucks; reduction of the total travelled distance; reduction of delivery times; minimization of distance travelled with minimal or no occupation of the truck; improving business processes of major importance; reducing personnel's occupation times; more flexible and efficient planning; improved communication and data, transfer of critical information for the whole enterprise; instant access to real-time data; etc.

References

[1] Archetti, C.; Mansini, R. and Speranza M. G.: The split delivery vehicle routing problem with small capacity: In Technical Report n. 201, Department of Quantitative Methods, University of Brescia, (2001).

[2] Azi, N., Gendreau, M. y Potvin, J.: An exact algorithm for a vehicle routing problem with time windows and multiple use of vehicles. In European Journal of Operational Research, Vol. 202, No. 3, pp 756-763, (2010).

[3] Balinzki, M. L. and Quandt, R. E.: On an Integer Program for a Delivery Problem:. In Operational Research, Vol. 12, No. 2, 1964, pp 300-304, (1964)

[4] Baptista, S.; Oliveira, R. C. and Zúquete, E.: A period vehicle routing case study. In European Journal of Operational Research, 139:220-229, Elsevier, (2002).

[5] Bard, J. F.; Huang, L.; Dror, M. and Jaillet, P.: A branch and cut algorithm for the VRP with satellite facilities. In IIE Transactions 30, pp. 821-834, (1997).

[6] Contardo, C.A.: Formulación y solución de un problema de ruteo de vehículos con demanda variable en tiempo real, trasbordos y ventanas de tiempo. In Memoria para optar al título de ingeniero civil matemático, Departamento de Ingeniería Matemática, Universidad de Chile, Santiago de Chile, Chile, (2005).

[7] Cordeau, J.-F.; Desaulniers, G.; Desrosiers, J.; Solomon, M. M. and Soumis, F.: VRP with time windows. In P. Toth and D. Vigo (eds.): The vehicle routing problem, SIAM Monographs on Discrete Mathematics and Applications, vol. 9, Philadelphia, PA, 157-193, (2002).

[8] CouchDB, Apache CouchDB project, http://couchdb.apache.org/, April 14, 2014.

[9] Dantzig G. B., Ramser J. H.: The truck dispatching problem. In Management Science, Vol. 6; pp 80–91, (1959).

[10] Dantzig G.B., Fulkerson R., Johnson S.: Solution of a large-scale travelling salesman problem. In Operations Research, Vol. 2, pp. 393– 410, (1954).

[11] Driver M, Valdes R, and Phifer G.: Rich Internet Applications are the next evolution of the Web. In Technical Report, Gartner, (2005).

[12] Dror, M.; Laporte G. and Trudeau P.: Vehicle routing with split deliveries. In Discrete Applied Mathematics 50, 239-254, (1994).

[13] Eksioglu B., Vural A. V., Reisman A.: The vehicle routing problem: A taxonomic review. In Computers & Indus-trial Engineering, Vol. 57 (4), pp. 1472-1483, (2009).

[14] Garvin, W. W.; Crandall, H. W.; John J.B. y Spellman, R. A.: Applications of Linear Programming in the Oil Industry. In Management Science, Vol. 3, 1957, pp 407, 1957.

[15] Hjorring, C.: The vehicle routing problem and local search metaheuristics. In PhD thesis, Chapter 2, Department of Engineering Science, The University of Auckland, (1995).

[16] Jacobs-Blecha, C. and Goetschalckx, M.: The vehicle routing problem with backhauls: properties and solution algorithms: In Technical Report, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia. Presented at the National Transportation Research Board, January 13-15, Washington DC, 1992.

[17] Laporte G.: Fifty years of vehicle routing. In Transportation Science, Vol. 43 (4), pp. 408–416, (2009).

[18] Laporte G., Osman I.H.: Routing problems: A bibliography. In Annals of Operations Research, Vol. 61, pp. 227–262, (1995).

[19] Laporte, G. and Louveaux, F. V.: Solving stochastic routing problems with the integer L-shaped method. In: Fleet Management and Logistics, T.G. Crainic and G. Laporte (eds.), 159-167, Kluwer Academic Publishers, Boston, (1998).

[20] LeafletJS, project website, http://leafletjs.com/, April 9, 2014.

[21] Luxen, D., & Sanders, P.: Hierarchy decomposition for faster user equilibria on road networks. In Experimental Algorithms (pp. 242-253). Springer Berlin Heidelberg, (2011).

[22] Meliá S., Gomez J.: The WebSA Approach: Applying Model Driven Engineering to Web Applications, Journal of Web Engineering, Vol. 5, No. 2, pp. 121-149, (2006).

[23] MongoDB, project website, https://www.mongodb.org/, April 10, 2014.

[24] MyBatis, Project webpage, http://blog.mybatis.org/, April 14, 2014.

[25] Nominatim, Project webpage, http://www.nominatim.org/, April 7, 2014.

[26] Olivera, A.: Heurísticas para problemas de ruteo de vehículos. In Reporte de investigación, Instituto de Computación – Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay, (2004).

[27] Open Street Maps, project web site, http://www.openstreetmap.org/, April 7, 2014.

[28] O'reilly, T.: What is Web 2.0: Design patterns and business models for the next generation of software. In Communications & strategies, 65, (2007).

[29] OSM Data format, http://wiki.openstreetmap.org/wiki/OSM_XML, April 7, 2014.

[30] PostgreSQL, http://www.postgresql.org/docs/9.3/static/datatype-json.html, April 9, 2014.

[31] Ralphs, T.; Hartman, J. and Galati, M.: Capacitated vehicle routing and some related problems. In Rutgers University, (2001).

[32] Righini, G.: Approximation algorithms for the vehicle routing problem with pick-up and delivery. In Note del Polo - Ricerca 33, Polo Didattico e di Ricerca di Crema, Università degli Studi di Milano, (2000).

[33] SmartGWT, Isomorphic software, http://www.smartclient.com/smartgwt/showcase/, April 21, 2014.