An Agent Based Travel Assistant for the Dutch Railway Network

Rothkrantz, Leon; Beelen, Martijn

DOI
10.1109/InfoTech58664.2023.10266886

Publication date
2023

Document Version
Final published version

Published in
2023 37th International Conference on Information Technologies, InfoTech 2023 - Proceedings

2023 International Conference on Information Technologies (InfoTech-2023), Proceedings of the IEEE Conference, Rec # 58664
20-21 September 2023, Bulgaria

An Agent Based Travel Assistant for the Dutch Railway Network

Leon Rothkrantz, Martijn Beelen
Faculty of Electronic Engineering, Mathematics and Computer Science, Delft University of Technology
Mekelweg 4, 2628 CD Delft, The Netherlands
Faculty of Transportation Sciences, Czech Technical University in Prague
l.j.m.rothkrantz@tudelft.nl

2023 International Conference on Information Technologies (InfoTech) | 979-8-3503-3805-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/INFOTECH58664.2023.10266886

Abstract – **For many years the Dutch NS Railway Company runs all trains in the Netherlands. It can be expected that next year different transport companies will exploit the Dutch railway network. Possible that they will use different systems to design timetables, generate travel plans and to handle delays. In this paper we research the coordination between different systems. A prototype of distributed systems in different areas has been developed using agent technology. The system has been tested in simulation studies.**

Keywords – **DYNET algorithm; intelligent agents; Jade platforms; multimodal transport; simulation.**

I. Introduction

Many researchers from the Dutch Research School for TRAnsport, Infrastructure and Logistics (TRAIL) were involved in multimodal transport research for many years. The goal was to design public transport routing systems, composed of different modalities without additional delay for the travelers. The system provides a travel plan from start to destination. On his way, the traveler will be informed about delays and will get an alternative routing plan if needed. The routing information will be displayed on a handheld as a Personal Intelligent Traveling Assistant (PITA).

The Dutch Company “Openbaar Vervoer Reisinformatie” (OVR) designed a website for travelers, to request a travel scheme from start to destination via the shortest route using in case different kind of transport modalities as trains, busses, trams, and fairies. The Company maintains the database of timetables and delays and provides access to the data via an App, enabling the traveler to plan its route. Because of the many customers, the system is not able to track all the users and provide personal support. OVR is a centralized system and is financial supported by other transport companies.

In the course of time different Transport Companies developed their own website and providing travelers access via special Apps. The idea of one central system maintaining different transport modalities is under discussion. It proved that different Transport Companies want to maintain and present their own time table data and upcoming delays. Some of them are not willing to support anymore a centralized route planning system mainly used by train travelers. There was also a need to include travelers using their cars, taxis or bikes, that is to say non-public transport. Many railway stations and bus station have huge parking lots. The inclusion of transport via the Dutch National Airport is also one of requirements.

In this paper we focus on the communication between different transport companies exploiting part of the Dutch railway system. The new providers may use different systems to design timetables and to handle delays. Some companies as Arriva exploit part of railway systems abroad and developed their own traveling systems. To test the cooperation between different systems, a test environment is needed, which will be discussed and tested in this paper.

A centralized information system enabling travelers to plan their route is the preferred option of Transport Companies. A decentralized distributed system has many advantages. The routing advice may be personalized and because the number of clients is less compared to a centralized system, it opens the option to support the traveler during his trip and in case of delays offer tailor-made alternatives. We developed a first prototype of a distributed, decentralized, agent-based system in [1]. But at those days OVR was afraid losing control of the travel systems market deploying a decentralized system. An update of the agent-based system will be presented and tested in this paper. To test the PITA system a simulated train environment was needed. We developed such an environment using the DYNET algorithm. That algorithm was also used by the DUTCH railway company for many years to design a routing system and a time table system for the trains.

The outline of the paper is as follows. In the next section we discuss related research. Then we present a model of the PITA system and discuss the advantages and disadvantages of a centralized and decentralized system. In the following section we present the design and implementation of the PITA system and we conclude this paper by a section Tests.

Fig. 1. Artistic impression of seamless multimodal transport (PITA)

II. Related works

In [1] we developed a first prototype of the PITA system. The system was tested in a special simulation environment. A special dialogue model was used [2]. That model was based on a corpus of 6.000 recorded dialogues in an OVR call center. Because of the risk OVR will lose control of providing information how to travel with public transport, the introduction of the system was postponed.

OVR started in 1991 with a call center to provide travel advices via public transport in the Netherlands [3]. The Travel planner includes timetables and delays of public transport companies exploiting trains, busses, trams and fairs. Step by step the service was extended via websites and Apps. At this moment the OVR database includes data from many transport companies as NS, Arriva, Keolis, Syntus, Breng, GVB, Connexxion, EBS, Hermes, Qbuzz, RET, HTM, Waterbus and U-OV.

In [4] we published a PITA system for car-drivers. The system was based on various technologies as Java and Open Street Map Api. The current traffic situation can be assessed by tracking car drivers provided with a smart phone including a GPS device. In the paper a system has been discussed to predict future delays in travelling time. A multi-agent approach for routing vehicle drivers was used based on historically recordings of traffic information.

In [5] Toma introduces a co-driver surveillance agent, which is able to monitor and evaluate the driving behavior of a car-driver. The driving behavior is measured by sensors in the car. In case the driver makes a mistake, the agent generates a warning alert. The knowledge of the reasoning system was based on analyzing driving behavior in a car simulator. The developed system has been tested in the same car simulator. Many co-drivers' system have been developed and will result finally in autonomous self-driving cars.

Aammari et all. researched in [6] the problem of a travel assistant able to adapt appointments of a car driver caused by delay of traffic jams. A car driver is assumed to notice his appointments in the calendar of Android. The virtual agent monitors the intensity of traffic. In case of delay the driver is alarmed about his time of the departure. The agent selects a fastest route using a special routing system based on recordings of historical traffic data and using a dynamic Dijkstra algorithm.

In [7] A. Sudhakar Reddy presents a chat bot as a virtual assistant. The chatbot was able to understand human speech and generates speech by synthetic voices. The system was trained on simple commands and used for management of emails, to-do lists and calendars with verbal commands. Similar systems are now available on the market. Well-known available virtual assistants are Siri, Alexa, AI Chatbot and of course end 2022 published ChatGPT.

In [8] van Uelsen presents a study on the perception of consumers of the usefulness of virtual assistants and the determination of variables, contributing to this perception. After a huge literature study on virtual assistants, he defines 7 hypotheses. These hypotheses are evaluated in an online survey. About 160 respondents took part in the survey. Perceived usefulness, perceived ease of use, relevance for the task, quality of virtual assistant was highly appreciated hypothesis.

Brill et al. researched in [9] the satisfaction of well-known digital assistants as Apple's Siri, Amazon's Alexa, Google's Google Assistant, Facebook Messenger assistant, Blackberry Assistant, Braina, Teneo, Speaktoit Assistant, Hound and Microsoft's Cortana. They analyzed survey responses from 244 users. It proved that the functional and topical use, varied by individuals. It proved that it was confirmed that expectations and confirmation of expectations have a positive and significant relationship on customer satisfaction with digital assistants. This study provides evidence that customer expectations are being satisfied through the digital assistant interaction experience.

In [10] Jagbeer et al. researched how to enhance the performance of a virtual assistant, AI assistants or digital assistants. Virtual assistants are able to perform a lot of tasks such as scheduling meetings, delivering messages, provide information on time tables and delivering news messages, according to a user profile. According to the authors, some virtual assistants as Google assistant and Cortana had their limitations and were not fully automated. As a testcase they researched handling songs, the forward and rewind function. One of the innovative ideas was the introduction of a virtual mouse that can be used for cursor control and clicking. The system gets input from a camera or finger clicking.

In [11] Ferreira et al. performed a survey study on virtual traveling agents. They demonstrated the effect of authenticity and attachment. When users view their communication as authentic, they become more engaged. Users more familiar to their intelligent virtual assistant in their thinking, feeling and using, are also more related to the information and recommendation provided by the device.

In [12] Khusnutdinov et al. present an open source and smart home system to develop Digital Personal Assistants (DPA) and a 3-rd party extension to show the functionality of DPA. The interface of the learning platform is based on natural language. A smart home system to control home appliances as light and temperature are used as a demonstrator.

In [13] Barcelo et al. present a literature overview of Intelligent Personal Assistants (IPA). They present IPA as an innovative interaction mode. The authors conclude that there is a vast literature on this topic, but no overview and analysis. The authors reviewed 3472 scientific articles published in the last six years. Next 58 most significant articles were selected and topics as architecture, security, privacy and usage were discussed. The authors conclude that usability, security, and privacy directly affect the confidence of the user in adopting an IPA.

In [14] Tulp researched the generation of time tables of trains. The developed DYNET algorithm provided the basis of the generation of timetables for the train network in the Netherlands for many years. This system is also implemented in our PITA simulation. Researchers from the TRAIL school developed an alternative system based on max-plus algebra to generate time tables.

In [15] Olsder et al. present a book on max-plus algebras as a framework to model discrete events systems, to describe the ordering and timing of events. The system enables the calculation of train time tables by mathematicians. The authors present discrete event systems in traffic systems, computer communication systems, production lines and flows in networks. After an introduction in max-plus algebra, the authors applied the theory to the design of timetables for railway networks. The authors conclude with stochastic systems as an extension of max-plus systems.

In [16, 17] Schrijver et al. represent his ideas about scheduling of trains on networks. He designed many models applied by the Dutch Railway company. At start trains and train compartments have to be distributed over the railway stations. A model has been used to transport travelers given the cohort of travelers and their preferred trips. Efficient circulation of rolling stock is an important research topic.

III. Personal Intelligent Traveling Assistant (PITA)

Before the introduction of digital train assistant systems, time tables of arriving and departing trains with corresponding platforms were visible at huge billboards at train stations. If a traveler has to switch trains, he has to monitor the information on the billboards with actual delays and move usual in haste to the platform. If he missed the train he has to go be back to the central arriving hall and consult the information on the billboard again. A less comfortable situation for a traveler in haste or with many suitcases. A digital solution was recommended and possible, because the actual train information was available in digital form before displaying on billboards.

The Dutch Railway company NS started with a website (see Figure 2). Via this website a traveler is able to plan his trip. He has to provide information for 4 out of 5 slots: departure station, arrival station, time of departure, time of arrival and date. The system computes the information for the fifth slot, usual the arrival time and the sequence of stations from begin to end and the stations where the traveler has to switch trains. The system generates a static advice. Unfortunately, a trip by train can be very dynamic caused by delays, incidents and bad weather conditions. Travelers in the train and waiting travelers on the platforms are informed over all possible delays and requested to update their trip.

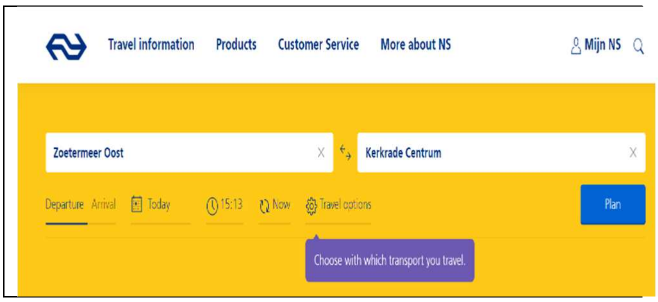

Fig. 2. Interface of the Railway NS website

Recently the NS-Railway company introduced a NS App (see Figure 3). This App enables the traveler to plan its trip and changes via his smart handheld. It is also possible to access the delay information at the central NS office. Unfortunately, it takes some time to find out the relevant information and adapt the trip to the changed situations. The reason is that information for all travelers is displayed and travelers are not able to filter out the relevant information.

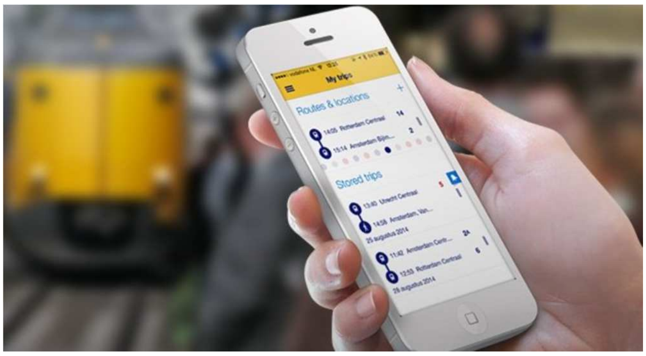

Fig. 3. Interface of the NS App

The goal of the PITA system is to generate a travel advice on request of the user. The user is routed from its starting point to its destination and in case of train incidents an adapted routing scheme is generated. The PITA system can be designed as a centralized or de-centralized system (see Figures 4 and 5). In a centralized approach an agent is monitoring all travelers. In a decentralized approach each traveler is represented by its own agent on his handheld computer.

In 2005 we designed our first prototype of a PITA system. To take a decision about a centralized or decentralized approach we organized a panel discussion. About 80 participants took part in the discussion selected from researchers from TRAIL, TUDelft, TNO and stakeholders from railway companies and OVR. The main conclusions from our panel discussion can be summarized as follows;

1. A centralized approach is designed for the whole cohort or travelers without personal adaptation. A tailor-made advice as in a de-centralized, personalized approach is not possible. That is the main reason, we are interested in a decentralized approach in our new PITA design.
2. In a centralized approach, a list of train delays and changes in the travel schemes, is send to all users. In a personalized decentralized approach, it is possible to focus on the travel scheme of the individual traveler and send information only if this is needed and tailor-made.
3. In a centralized approach, it is usual not possible to handle requests for additional information of individual travelers. In the personalized decentralized approach, tailor made information is provided and less requests for additional information can be expected.
4. In a personalized, decentralized approach it is possible to take care of individual wishes. For example, some travelers prefer a trip with a minimal of train changes because they have a lot of heavy baggage on their way to Schiphol Airport.
5. In a personal decentralized approach, the system saves given advices and is able to adapt given advices.
6. In case of delay of a train, the PITA system has to inform travelers suffering from this delay. This are travelers in the delayed train and travelers waiting on arrival stations. These travelers can be filtered out from the database of all trip planning.

IV. ARCHITECTURE OF A (DE-) CENTRALIZED PITA-SYSTEM

In Figures 4 and 5 we display the architectures of a centralized and decentralized version of the PITA. As expected in the centralized approach the route planning system, database and database of delays is running on a central server. Clients have access to the central route planning system via their handheld devices. Via a dialogue manager clients can have contact with the central agent. The central agent can access the route planning system running on the central server. Also, the database with delay information is running on the central server.

A great advantage of a central system is that the maintenance of the system is relatively easy. As soon as we have a distributed system as in the decentralized approach, we have the problem of synchronization and update of different components. But a great disadvantage of a centralized system is that the whole system fails if the central server cracks down. Decentralized systems keep on running in case of central system failure, but the update and synchronization may be delayed. Most handheld devices are nowadays so powerful that it is possible to run the route planning system on these handhelds and also store copies of relevant data and delay data on these handhelds. But then it is necessary to update data on all these handhelds and synchronize the communication between them. As a consequence, there is a lot of message-passing between different components.

The great advantage of a decentralized system is that it is possible to compute individual route planning system, store these planning in the database and track individuals and send alerts for adaptation in case for example of delays. In a decentralized system, it is easy to store user preferences and compute a user profile based on their travel history.

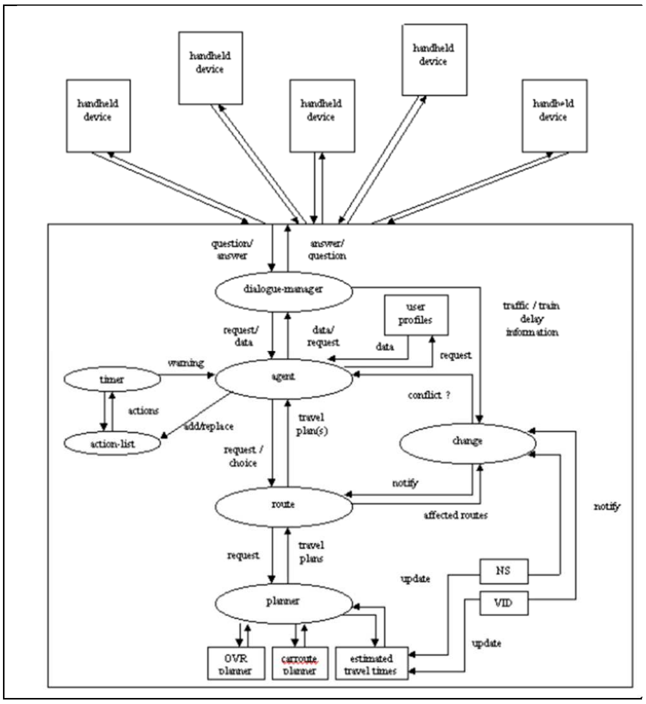

Fig. 4. Architecture of the PITA system (centralized system)

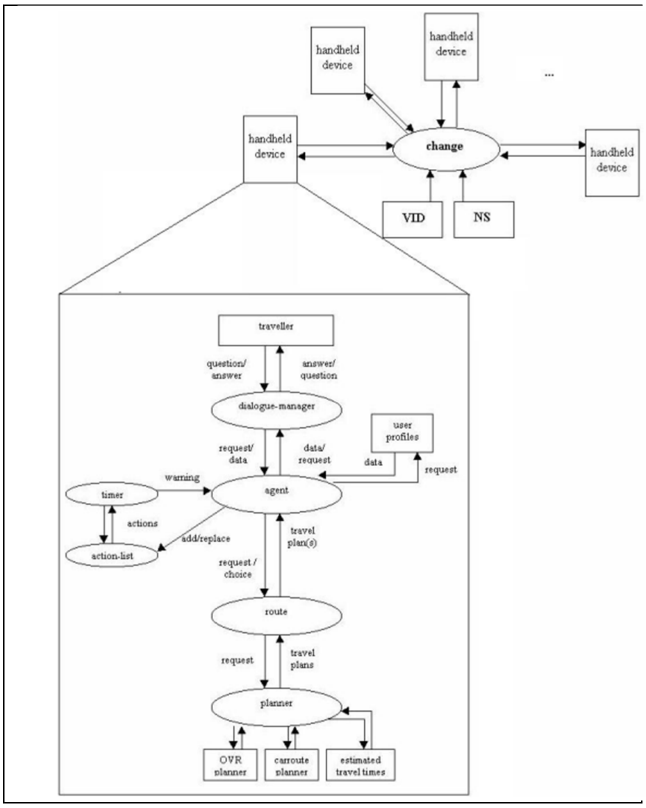

Fig. 5. Architecture of the PITA system (de-centralized system)

V. SYSTEM DESIGN

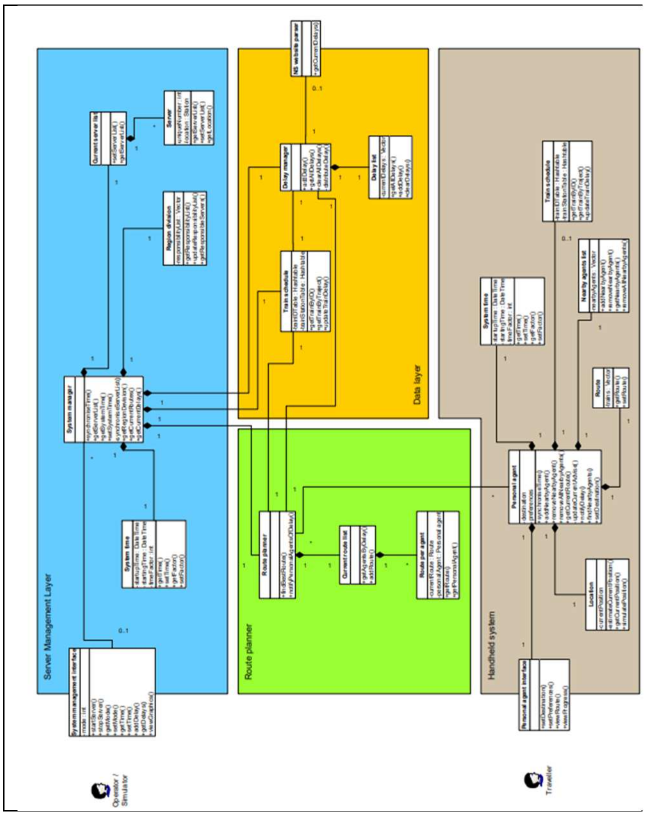

Fig. 6. Class diagram of the PITA system

In Figure 4 and 5, we displayed the architecture of a centralized and decentralized system. Our goal is to implement a prototype for both systems and to test both systems in our simulation environment. As a first step we designed a System Object Model (SOM). We used Unified Modeling Language (UML) to visualize the main actors and classes. Such a UML diagram provides better understanding of a system.

Our distributed agent model is composed of a system of distributed servers. These servers have to synchronize and update the whole system. To enable the synchronization, each server has a main component, a System Manager. That module is responsible for updating and synchronization of the time, and possible delays. In case more companies are exploiting the railway system every region has his central server. Optimal communication between these servers is of vital importance for the functioning of the whole system. System Managers and Operators can have a full overview of the system via the distributed System Manager module and in case is able to change different settings.

The personal agent keeps track of delays and is able to update the route taking care of the request of the traveler and his requirements. Updating the time tables and upcoming delays is one the main functions of the PITA system. The static route information from the start has to be updated in case of delays and communicated between the different components of the PITA system.

To model the information flow between different objects we designed sequence diagrams as displayed in Figure 7,8 and 9. These sequence diagrams are a visualization of the information exchange between the different objects.

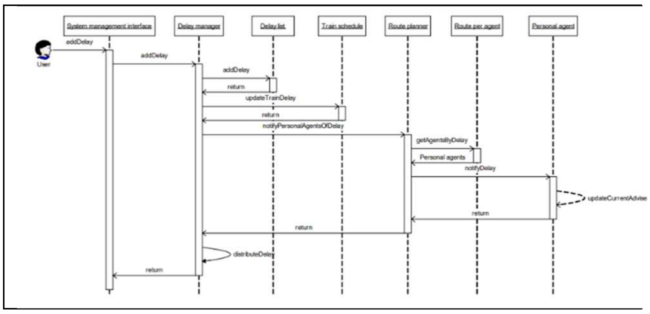

Fig. 7. Implementation of new delay visualized in Sequence diagram

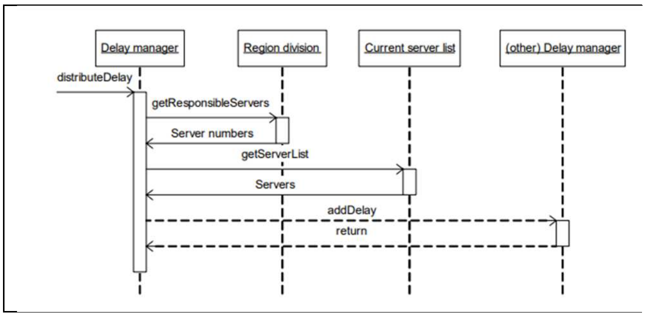

Fig. 8. Distribution of delays over different servers visualized in a Sequence diagram

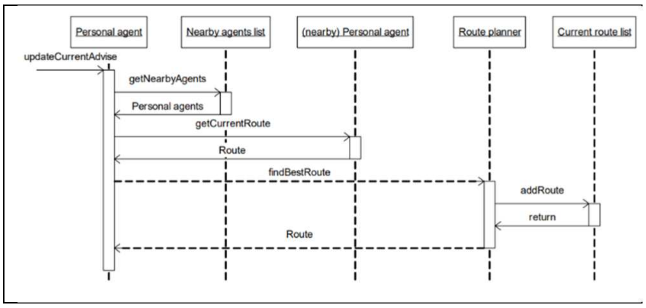

Fig. 9. Updating travel advice visualized in sequence diagram

VI. The Dynet algorithm

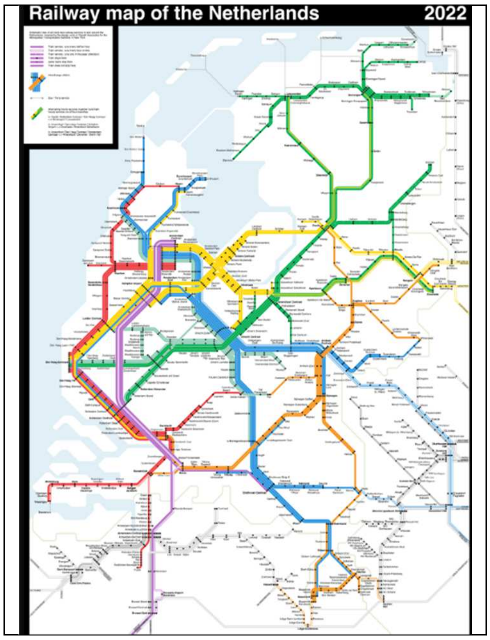

Fig. 10. Railway map of The Netherlands

In the Netherlands there is a very dense network of railway tracks and stations. There are single, double or even multi- tracks. We have transport of persons or freight transport by rail. There is a fast train network with many train compartments and a network of local stop trains. Most trains are electrified, except some local trains. The Railway network should be used to transport travelers in an efficient and effective way. As expected in the Western part with a high population density and big cities, there will run more trains than in the Northern part for example. But for societal and political reasons there are daily trains between most parts of the Netherlands. In the Western part trains run every 15 minutes between the big cities. All these requirements set constraints for a route planning system. Some of these constraints are contradictory. Most people want to travel in the shortest time from A to B. Other people accept a longer travel time if they have less switches between trains. In the evening there run less trains, but still needed for people without cars. But the strongest constraint is that there will be trains from and to the least populated areas of the Netherlands.

At this moment there are many route planning systems available. Tulp and co [Tulp91] designed DYNET, a route planning system for trains in 1991for the Netherlands and this system has been used for many years. We also implemented a version in our railway map simulation system.

As a first step we have to define the frequency of trains between different stations, and starting and arrival time, mainly based on political/societal reasons. Freight transport takes mainly place over night or via special railway tracks. Starting trains should be available at the starting point, compartments distributed over the country [16, 17]. This requires sometimes that empty trains are transported over night to the starting point. In the TULP algorithm first the fast trains are planned and next local trains.

The algorithm of Tulp is composed of a forward and backward pass. In Figure 11, 12 and 13 we show the successive steps of the algorithm.

Consider a discrete dynamic network consisting of the graph $G = (V,E)$ and the connection function *CON*. The maximum value of *CON* at a vertex *u* is *maxiCON(u)*. The number of train changes in a path *P* is denoted by *CHANGES(P)*. The *change_value* is the time that we are prepared to travel to avoid one train change. The corrected *end* value of a path *P*, $c_{end}(P)$, and the corrected *start* value of a path *P*, $c_{start}(P)$, are defined as follows:

$c_{end}(P) = end(P) + CHANGES(P) \times change\_value$

$c_{start}(P) = start(P) - CHANGES(P) \times change\_value$

continued on the next page…

Fig. 11. The DYNET algorithm (part1)

The label of a vertex *v* is $\lambda(v)$ in the forward pass and $k(v)$ in the backward pass. The smallest end value that was tried to label *u* from *v* is denoted by $\omega(u,v)$. The number of train changes of the (corrected) best path arriving at a vertex *v* is denoted by $\varphi(v)$. A train riding a sequence of edges e_0, e_1, e_2 is identified by an extra edge attribute $id(e_0) = id(e_1) = id(e_2)$. The two special vertices *s* and *t* of the network are the starting and terminating vertices. We want to find a legal path from *s* to *t* in our discrete dynamic network, where the corrected *end* value of the path is minimum and given this corrected *end* value, the corrected *start* value of the path is maximum and its value is at least T_{start}.

Then the DYNET algorithm, consisting of two passes, is as follows:

PASS 1:

(1) $\lambda(s) \leftarrow Tstart$ and for all $v \in V$, $v \neq s$, $\lambda(v) \leftarrow \infty$.
Create a partial path P_0 consisting of *s* only, $c_{end}(P_0) \leftarrow T_{start}$.
For all $v \in V$, $\omega(u,v) = \infty$ for each neighbour *u* of *v*.

(2) $F \leftarrow \{ P_0 \}$.

(3) Let P_m be a partial path $s, e_0, u_1, .. , u_{j-1}, e_{j-1}, u_j$ in *F* for which $c_{end}(P_m)$ is minimum; if *F* is empty then stop, no complete path could be found.

(4) if $u_j = t$, stop, P_m is a complete path with an optimal corrected *end* value.

(5) if $CHANGES(P_m) = \varphi(u_j)$ and $c_{end}(P_m) < \lambda(u_j) + maxiCON(u_j)$, then for every relevant edge $e_j : u_j \rightarrow u_{j+1}$:
create a partial path $P_n = s, e_0, u_1, .. , u_{j-1}, e_{j-1}, u_j, e_j, u_{j+1}$.
$c_{end}(P_n) \leftarrow end(P_n) + CHANGES(P_n) \times change\_value$
if $\lambda(u_{j+1}) > c_{end}(P_n)$
then $\lambda(u_{j+1}) \leftarrow c_{end}(P_n)$ and $\varphi(u_{j+1}) \leftarrow CHANGES(P_n)$.
$F \leftarrow F + \{ P_n \}$.
if $end(e_j) < \omega(u_{j+1}, u_j)$
then $\omega(u_{j+1}, u_j) \leftarrow end(e_j)$.

(6) if $CHANGES(P_m) = \varphi(u_j)$ and $\lambda(u_j) + maxiCON(u_j) \leq c_{end}(P_m)$.
then for the edge $e_j : u_j \rightarrow u_{j+1}$ for which $id(e_j) = id(e_{j-1})$:
create a partial path $P_n = s, e_0, u_1, .. , u_{j-1}, e_{j-1}, u_j, e_j, u_{j+1}$.
$c_{end}(P_n) \leftarrow end(P_n) + CHANGES(P_n) \times change\_value$
if $\lambda(u_{j+1}) > c_{end}(P_n)$
then $\lambda(u_{j+1}) \leftarrow c_{end}(P_n)$ and $\varphi(u_{j+1}) \leftarrow CHANGES(P_n)$.
$F \leftarrow F + \{ P_n \}$.

(7) $F \leftarrow F - \{ P_m \}$ and go to step (3).

A relevant edge is defined as follows: given a partial path $u_0, .. , u_{j-1}, e_{j-1}, u_j$ and a vertex u_{j+1}, then the relevant edges from u_j to u_{j+1} are the edges $e_j : u_j \rightarrow u_{j+1}$ for which the following two *ordered* conditions hold:
(1) $start(e_j) \geq end(e_{j-1}) + CON(u_j, e_{j-1}, e_j)$, and
(2) $end(e_j) < end(e_{min}) + maxiCON(u_{j+1})$,
where $end(e_{min})$ is the minimum *end* value of any edge satisfying (1).

continued on the next page…

Fig. 12. The DYNET algorithm (part2)

PASS 2:

(1) $k(t) \leftarrow \lambda(t)$ and for all $v \in V$, $v \neq t$, $k(v) \leftarrow -\infty$.
Create a partial path P_0 consisting of *t* only, $c_{start}(P_0) \leftarrow \lambda(t)$.

(2) $F \leftarrow \{ P_0 \}$.

(3) Let P_m be a partial path $u_j, .. , u_{k-1}, e_{k-1}, t$ in *F* for which $c_{start}(P_m)$ is maximum.

(4) if $u_j = s$, stop, P_m is an optimal complete path.

(5) if $\varphi(u_j) = CHANGES(P_m)$ and $c_{start}(P_m) > k(u_j) - maxiCON(u_j)$, then for every relevant edge $e_{j-1} : u_{j-1} \rightarrow u_j$, with $\omega(u_j, u_{j-1}) \leq start(P_m)$:
create a partial path $P_n = u_{j-1}, e_{j-1}, u_j, .. , u_{k-1}, e_{k-1}, t$.
$c_{start}(P_n) \leftarrow start(P_n) - CHANGES(P_n) \times change\_value$
if $k(u_{j-1}) < c_{start}(P_n)$
then $k(u_{j-1}) \leftarrow c_{start}(P_n)$ and $\varphi(u_{j-1}) \leftarrow CHANGES(P_n)$.
$F \leftarrow F + \{ P_n \}$.

(6) if $\varphi(u_j) = CHANGES(P_m)$ and $k(u_j) - maxiCON(u_j) > c_{start}(P_m)$.
then for the edge $e_{j-1} : u_{j-1} \rightarrow u_j$, with $id(e_{j-1}) = id(e_j)$,
and with $\omega(u_j, u_{j-1}) \leq start(P_m)$,
create a partial path $P_n = u_{j-1}, e_{j-1}, u_j, .. , u_{k-1}, e_{k-1}, t$.
$c_{start}(P_n) \leftarrow start(P_n) - CHANGES(P_n) \times change\_value$
if $k(u_{j-1}) < c_{start}(P_n)$
then $k(u_{j-1}) \leftarrow c_{start}(P_n)$ and $\varphi(u_{j-1}) \leftarrow CHANGES(P_n)$.
$F \leftarrow F + \{ P_n \}$.

(7) $F \leftarrow F - \{ P_m \}$ and go to step (3).

A relevant edge is defined as follows: given a partial path $u_j, e_j, u_{j+1}, .. , u_k$ and a vertex u_{j-1}, then the relevant edges from u_{j-1} to u_j are the edges $e_{j-1} : u_{j-1} \rightarrow u_j$ for which the following two *ordered* conditions hold:
(1) $end(e_{j-1}) \leq start(e_j) - CON(u_j, e_{j-1}, e_j)$, and
(2) $start(e_{j-1}) > start(e_{max}) - maxiCON(u_{j-1})$,
where $start(e_{max})$ is the maximum *start* value of any edge satisfying (1).

Fig. 13. The DYNET algorithm (part3)

VII. The Distributed Agent Platform

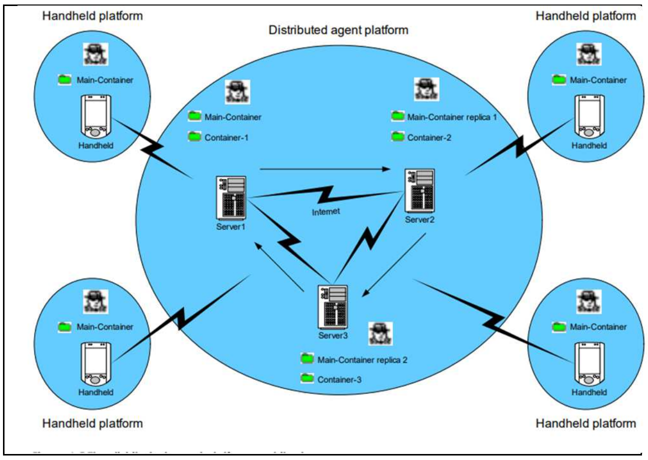

Fig. 14. The distributed agent platform

Our distributed agent platform as displayed in Figure 14, is composed of many components. First, we have a network of connected main-containers. These containers store the routing data and delay data. The information on these containers is updated regularly to keep the distributed data consistent. In case one container crashed, it can be updated by info from other containers. The various handheld platforms are connected with the main containers as user interface in the centralized system or subcomponent in the decentralized version. In the last case a copy of the routing algorithm and database is running on the handheld platform. As stated before, this requires a careful update and synchronization mechanism.

The agent model allows to model a handheld as an agent. An agent is a relative autonomous entity able to perceive its environment and take appropriate actions. To model users as agents enable them to take care of their travel destination, their preferences and supervision of the traveler on its route. In a centralized model it is almost impossible to take care of thousands of travelers in parallel. But the main advantage of the agent-based model is that if the central computer crashed the regional servers continue working and start updating.

VIII. Tests

The time table, train routing system has been designed using DYNET and is extensively tested in the past years. On the DUTCH rail network different type of trains are running such as fast trains, InterCitys but also stop trains. The departure and arrival of these trains is according to the time table systems. When there are more providers on the rail network, the set of stations and rails is divided among all these providers. At this moment most trains are exploited by NS. Only regional trains are exploited by local companies. Travelers have to change trains from the main network to regional trains. This is needed because regional trains run only in the regions. But for convenience of the travelers is the timetable system of regional trains included and adapted to the whole system. If in the future other companies than NS will be involved in the main part of the network and run trains. It is not recommended that travelers have to change trains.

Our simulation environment enables test experiments. Our focus of interest is the smooth connections between part of the railway network exploited by different companies. If a traveler has a regional destination, he has to switch trains at some station. Regional trains are less frequent than trains on the main network so it is important to catch a connecting regional train in time. Fortunately, the regional planning system adapts dynamically to the regional planning system. In case a train on the main network is delayed, connecting regional trains adapt their departure time as much as possible. Thanks to the decentralized PITA system travelers will be informed about changes in their routes. In Figure 15 we display the rail network. Stations are presented as colored dots. Stations exploited by different companies have their own specific colors.

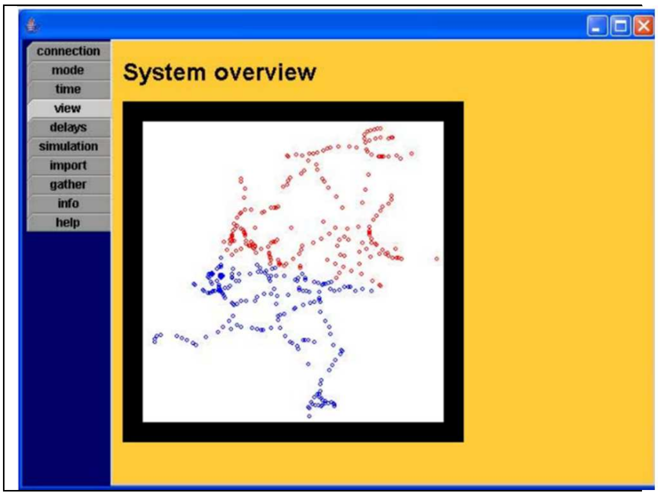

Fig. 15. The region division between two different servers

In a first experiment we tested a trip from a student from the University of Maastricht, located in the southern region of the Netherlands to the University of Groningen located in the Northern part of the Netherlands. The Intercity train Maastricht-Groningen runs usually strictly on time, so it is important to arrive timely at the railway station of Maastricht. In the time table an additional buffer of 10 minutes has been reserved for regional trains to Maastricht. Short delays in the region Maastricht are no problem. It proves that this buffer is needed in case of bad weather, shortage of operators caused by epidemic diseases and material problems caused by saving in maintenance. As a backup bus transport has been used and travelers are advised to switch in modality.

At start the traveler has to define his stations of departure and arrival. The PITA system assumes that the traveler wants to travel today immediately in the shortest time and preferred the shortest route. In case the traveler makes this trip regularly, this route and corresponding preferences are stored in his personal database. At the switching station Groningen, the PITA system sends a request to the regional server that a traveler wants to travel from Groningen to Groningen Noord. If possible and needed in case of many requests the local train waits for some minutes if needed. On request the traveler may be informed about the progress of the travel. The route is visualized in Figure 15 and information about possible delays and changed arrival time is presented as displayed in Figure 16. This info enables travelers to take actions in time.

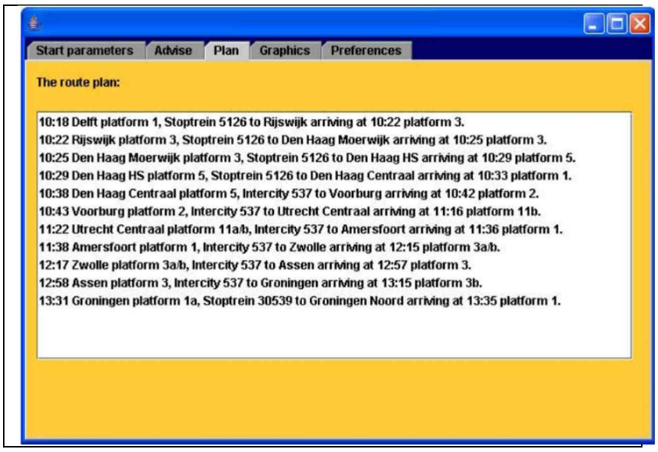

Fig. 16. Information about route plan

Thanks to our decentralized agent-based approach local delays are handled locally. This result in less overload of the central managers in case of delay problems at many locations. From the other site it is not possible to take central measures with a great region impact. The autonomous character of the agents enables local solutions. And hopefully all local solutions guarantee global solutions, which is unproven yet. From survey studies it proves that traveler prefer to have control of their routing plans, but they realize that measurements for the whole cohort of travelers are needed.

A great advantage of the PITA systems is that travelers and managers can be informed about the ongoing situation quite easily. Visual observations are communicated via the PITA system. Travelers are familiar to this way of communication by the regular use of social media. Aggression in trains, harassment, pickpockets, travelers without tickets are observed by travelers in the compartments and reported via PITA. The NS train company will be alarmed and take measurements for the next train stop. For many years intelligent surveillance cameras were installed in trains. But because of privacy reasons they have been removed again, but the PITA system may provide an alternative solution.

IX. Conclusions

For many years The Dutch railway company NS was the only company exploiting the network of Dutch Railways, transporting travelers from starting station to arrival station. Some years ago, a second company Arriva exploits a minor part of the railway network in the North-East part of the Netherlands. But it may be expected that coming years many more companies will join the exploitation of the Dutch railway market. Probably these new providers will use their own travel planners, routing and delay system. In this paper we researched the compatibility of different systems by designing a simulation environment with different routing and planning systems. Our new simulation environment was designed, using a distributed agent technology. It proved it was possible to implement different systems on different servers in parallel, taking care of the compatibility of those systems.

The systems were tested by virtual agents traveling from randomly selected departure station to arrival station. By introducing upcoming delays, we tested the resilience of the system. The Dutch railway network has a high occupancy rate. As a consequence, a short delay has usually a great impact. The coming years a high priority is given to the design and implementation of new railway tracks. Before eventually introduction these tracks have to be tested. Our simulation environment enables necessary experiments.

The personal Intelligent Traveling Assistant (PITA) was successful emulated. It is possible to implement hard coded user preferences, such as a list of regular travels and a limited set of stations used as starting point or end point. It is also possible to set the preferred modality, such as text or speech. The user is able to install his own preferences.

From survey studies it proved that PITA like system were preferred by travelers. Travelers like to have a PITA system working in the background, taking care of all upcoming messages and delays. In principle a traveler prefers to be in charge, but the traveler realized that he has only partly control of the communication between agents.

References

[1] L. Rothkrantz, D. Datcu and M. Beelen. “Personal Intelligent Travel Assistant. A distributed approach.” In Proceedings of the 2005 International Conference on Artificial Intelligence, ICAI 2005, Las Vegas, Nevada, USA, June 27-30, 2005, Volume 1.

[2] R. J. van Vark, J. P. M. de Vreught and L. J. M. Rothkrantz. “Classification of public Transport Information Dialogues using an information based coding scheme.” Workshop on Dialoge Processing in Spoken Language Systems, 1996, pp. 55-69.

[3] https://9292.nl
https://www.anwb.nl › Routeplanner
https://www.arriva.nl › consumenten › reisinformatie-1
https://www.connexxion.nl › actuele-reisinformatie ›
https://www.zuid-holland.nl › nieuwe-dienst. regeling-ov.

[4] A. Radu, L. Rothkrantz and M. Novak. “Digital Traveler Assistant.” Informatics in Control, Automation and Robotics, LNEE 174, pp. 101–114.

[5] M. Toma, Mirela Popa, Leon Rothkrantz. “An Intelligent Co-Driver Surveillance System.” Acta Polytechnica CTU Proceedings 12, pp. 83–93, 2017.

[6] L. Aammari, M. Hazebroek, J. Isling and R. Tjinasjoe. “The Keen Android Travel Extension (KATE)”. Repository TUDelft, 2011.

[7] A. Sudhakar, Reddy M. Vishwa, C. Raju Kumar, and Saumya. “Virtual Assistant Using Artificial Intelligence.” JETIR March 2020, Volume 7, Issue 3.

[8] Alexander Carl Alfred van Uelsen. “Usefulness vs. Uncanniness: Exploring the perceived usefulness of virtual assistants in the customer journey.” Report University of Twente ,The Netherlands. 2021 pp. 1-21.

[9] T. Brill, L. Munoz, R. Miller. “Siri, Alexa, and other digital assistants: a study of customer satisfaction with artificial intelligence applications Journal of Marketing Management.” 2019, 1472-1476.

[10] J. Singh, Y. Goel, S. Jain, S. Yada. “Virtual Mouse and Assistant: A Technological Revolution Of Artificial Intelligence. Journal of Pharmaceutical Negative Results, Volume 13, Special Issue 10, 2022.

[11] M. Ferreira, S. Correia Loureiro, M. Ashfaq and H. Pereira. “Tourist-virtual assistant communication: the power of attachment and authenticity.” Anatolia: An International Journal of Tourism and Hospitality Research. 2022.

[12] A. Khusnutdinov, D. Usachev, M. Mazzara, A. Khan, I. Panchenko. “Open-source platform Digital Personal Assistant. [1801.03650] Open-source platform Digital Personal Assistant (arxiv.org)

[13] A. de Barcelos Silva , M. Gomes , C. da Costa , R. da Rosa Righi, J. Barbosa, G. Pessin , G. Doncker , G. Federizzi. “Intelligent personal assistants: A systematic literature review.” Expert Systems with Applications Volume 147, 1 June 2020, 113193.

[14] E. Tulp, Searching Time-table Networks, PhD thesis, Vrije Universiteit van Amsterdam, Drukkerij Elinkwijk BV, Utrecht, oktober 31, 1991.

[15] B. Heidergott, G.J. Olsder, J. van der Woude. Maxplus algebra and its applications. Princeton University Press.

[16] A. Alfieri, R. Groot, L. Kroon, A. Schrijver. “ Efficient circulation of railway rolling stock.“ Transportation Science 40 (2006) 378—391.

[17] P.J. Fioole, L.G. Kroon, G. Maróti, A. Schrijver. “ A rolling stock circulation model for combining and splitting of passenger trains.“ European Journal of Operational Research 174 (2006) 1281--1297.