Service-controlled agile logistics

J.T.W. Damen

The author

J.T.W. Damen is a Consultant with KPN Research, Leidschendam, The Netherlands.

Keywords

Logistics, Distribution, Channel management

Abstract

Current logistics systems are unable to react fast enough to rapidly changing environments, mainly because they are heavily focused on the goods handling processes. To compensate for this lack of flexibility, logistics services should be carried out by independently controlled logistics resources. But these resources together do have to guarantee the overall quality of services. Service-controlled agile logistics solves the conflict that arises – independent resources working together – by strictly distinguishing between control and handling. It is based on control of logistics processes by the requested services themselves, which create their own "agents", made responsible for realizing the service in the best possible way under changing circumstances. In order to examine the feasibility of this approach, a simulation program has been developed, and some preliminary results are presented.

Electronic access

The research register for this journal is available at **http://www.mcbup.com/research_registers**

The current issue and full text archive of this journal is available at **http://www.emerald-library.com/ft**

Logistics Information Management
Volume 14 · Number 3 · 2001 · pp. 185–195

Introduction

The standard approach to deliver logistics express services has been (and still is) to let the client make a choice from a limited set of alternatives, mainly differentiating in time and cost. For instance, a major European express service provider offers "Euro services", with express documents, next-day parcels, next-day freight, two- to three-day or four-day parcels and freight, and "World services", with express documents, express parcels and express freight (TNT, 1998). Depending on the client's choice, a default path through the logistics network is chosen, set beforehand for each possibility. Necessary transport and handling capacities are estimated ahead based on experience from the past. A tight scheduling of transports is at the kernel of such a logistics servicing system.

In the world of distribution and express logistics, this is no longer sufficient. Clients demand more variety, and better control. In an earlier study, Damen (1994) has shown that the "old" approach is inherently inflexible. It has been recognized, that the standard way of logistics control is more and more unable to manage the dynamics of the logistics service requests of today (de Kok, 1998). As an example, the rapidly increasing use of e-commerce requires a new approach; agility in logistics is needed.

In the area of manufacturing, the need for agility has already been recognized for some time. Agile manufacturing can be defined as the capability to survive in an environment of continuous and unpredictable change by reacting quickly and effectively to changing markets, driven by customer-designed products and services (Gunasekaran, 1998). The aim in agile manufacturing is to develop the agility to form virtual corporations rapidly so that changes in markets and windows of opportunity can be exploited (Kidd, 1994). It is also recognized, that the role of logistics is gradually changing from a distinct support function to a core business function. In order to provide speed to market and fast, flexible customer response, a new logistics environment is needed. "Enterprise logistics" has three key properties:

(1) the ability to provide a seamless delivery process;
(2) the ability to track materials as they are in transit; and

Logistics Information Management
Volume 14 · Number 3 · 2001 · 185–195

(3) the ability to adjust transportation agreements based on specific customer demands (Greis and Kasarda, 1997).

In the literature, less can be found about agile logistics in general, outside or between manufacturing companies. Narasimhan and Das (1999) investigated the correlation between supply chain management practices and manufacturing agility. Taylor (1995) merely states, that agile logistics is the efficient movement of material throughout an organization in response to market demand. Others have sought improvements in this field of logistics in better cooperation between organizations (Babeliowsky, 1997; Sheombar, 1995; CrossFlow Project, 1998), making use of modern information technology or to improve control by introducing better planning systems (Hoogeweegen, 1997; Verwijmeren, 1998).

The route we follow to find a solution for managing the dynamics of modern logistics is close to that of Evers (1998) and Evers *et al.* (1999), and can be characterized by the statement of the Global Logistics Research Initiative (GLORI):

> Our mission is to develop new tools for managing the increasing complexity and turbulence in today's marketplace (GLORI, 1999).

Our approach has been stimulated by arguments from three different directions:

(1) control of a flexible logistics system should be completely information driven;
(2) logistics service providing should be more client friendly; and
(3) modern IT tools should be used to boost logistics control.

We developed service-controlled agile logistics (SCAL) as a new model for a logistics control system, that fulfils all three statements. It is an object-oriented model, based on developments in distributed object-oriented technology. In a system according to this model, each requested service is treated as an individual entity, for which a temporary path through the logistics network is planned, built and kept alive during execution by a dedicated "agent". To examine the feasibility of this approach, we used simulation. We implemented an experimental discrete event simulation program, including a prototype logistics control system, the first stage of which has been completed so far. We showed the good working by simulating a European wide express logistics network, based on the new model. We simulated both, the "old" days logistics control system and the proposed new system, and we compared – fairly roughly – the results.

In the next sections, we will describe the principles of a SCAL system, the setup of the experimental package, the results until now and how the future looks. We first focus on the two layers of an agile logistics system, representing the physical handling and information processing parts. Then, we describe the object model the system is based on with its two most active parts: planning and scheduling. Next, the prototype will be described, followed by some early results. Before finishing with conclusions, future perspectives are discussed.

Agile logistics system

Two main trends in the world of logistics can be distinguished. First, logistics services are requested for larger distances, from national to international to intercontinental to world-wide. Second, the dynamical aspects of the requests tend to increase. Lead times become shorter. Service requests vary over broader ranges, and change more frequently and more rapidly. Alterations even during execution become no exception any more.

Both trends have one factor in common: use of independent parties. Covering greater distances demands more parties to be involved, together reaching longer ranges. In an earlier study, Damen (1994) showed that flexibility in logistics systems is obtained by reverting to the basic (physical) processes that are controlled as independent entities. Logistics services can be divided into small "chunks" of operations for which often several alternatives are available. Responsiveness to changing circumstances is obtained by choosing the right alternative at the right time. With smaller "chunks", more alternatives exist and the range of available choices is wider. This again refers to making use of more, independently controlled parties. Together, they have to realize complete logistics services – we have called this "integrated logistics service providing" (see also IDEAS, 1994).

These parties should respond fast to changing requests. As said, this is easier if parties are only involved in small parts of the

Logistics Information Management
Volume 14 · Number 3 · 2001 · 185–195

services requested for. But commonly, logistics service providers try to cover service requests as a whole. One stop shopping is the adagio. Since their systems heavily lean on fixed chains of physical goods handling processes, they are mainly unable to react quickly to fast changing environments. Flexibility and fast reaction depend on fast decision making, which is typically an information processing phenomenon. Control belongs to the information processing domain, not the goods handling domain. Decoupling information processing from goods handling is necessary for being able to react fast.

But then another question arises: when services are executed by several parties each doing just a small part, who is going to guarantee the overall quality? This closely relates to how logistics services are made more client friendly. SCAL gives a simple answer: let logistics be controlled by the services themselves! It redirects the decision making back to where it belongs: the services requested for. They "know" at best what should be done, how "they" should be executed. In SCAL, each requested service is treated as an autonomous entity. It creates its own "agent", that is made responsible for realizing what has been requested for, in the best possible way under changing circumstances.

Typically, logistics resources that can realize (parts of) services, such as transports, sorting centres, warehouses, docking, undocking, are geographically distributed. We can assume that it is always possible to have goods travel from any (relevant) point in the world to any other. All these resources together constitute one large goods handling network, the logistics network. This is a physical network, distributed over the world globe, and populated by a huge amount of players in the field.

Every single logistics service is defined by a time-bound path (or paths, if the service concerns several distinct goods) through this network. The service agent must be able to visit several resources in the logistics network in order to search for the best path under temporal conditions, and to make agreements upon necessary logistics actions. An information network is needed, that has connections to the logistics network at places where resources are located and controlled.

Therefore, a SCAL system consists of two main layers:

(1) an information network for the service agents to travel, visit resources and follow the execution of the service requirements; and
(2) a logistics network for transporting and processing physical entities.

The goods are the common factor, the objects both layers are dealing with.

Two different concepts "rule" the two layers: at the information layer, *services* are the main entities, while *resources* play the main role within the goods processing layer. Their interests, however, conflict. In general, services request that goods are processed immediately, as soon as possible or exactly on time; they do not worry about other services. For instance, if a parcel has to be transported, the truck should start the very moment the parcel is loaded. To the contrary, logistics operators invest in resources to make money out of their use and therefore, again generally speaking, they try to gather as many goods as possible for processing in order to fully utilize the available capacity. They do not worry about other resources. For instance, a parcel transport would wait and delay all parcels already loaded until the truck is full.

When logistics service providing (the resources side) concentrates on the processes, and when requested services involve larger geographical areas and greater dynamics, this conflict becomes the main obstacle for quality guaranteed logistics control. SCAL transforms this conflict into the hot spot of logistics control, by re-establishing the ultimate functions of the two layers, goods handling and decision making, and making it the arena of negotiation for service requesters and resource providers.

Model

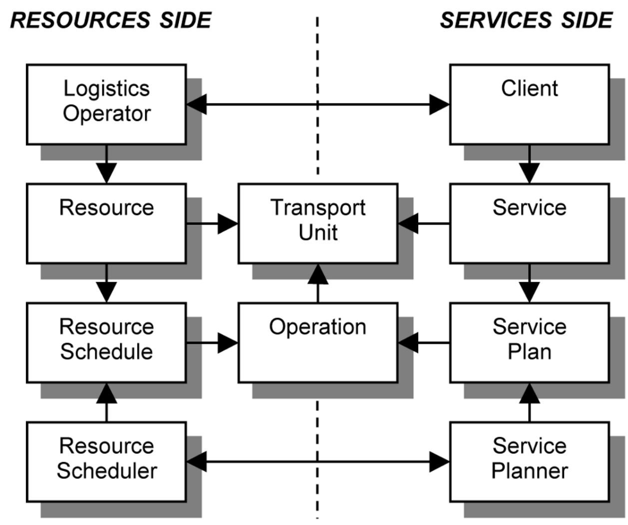
For the sake of clarity, and to sharpen the distinction between the two layers, the following terminology will be used.

Terminology

- *Client*: someone (or probably an organization, a higher order process) requesting a logistics service. The client initiates services, is interested in the realization in accordance with his requirements and is willing to pay for it.

- *Logistics operator*: someone (or an organization) who has invested in logistics tools and equipment. The logistics operator creates resources and is interested in making money out of them in accordance with his requirements.
- *Service*: a set of coherent logistics activities (seen both as a product and as a process) required to achieve a defined goal in space and time concerning some physical goods, especially those requiring transportation over non trivial, geographical distances.
- *Resource*: tool or equipment that carries out logistics activities with defined capacity in volume and in time, bound to a certain geographical area.
- *Planning*: defining the way a certain requested service will be realized, what resources will be used at what time and capacity.
- *Scheduling*: defining the way a certain resource will be utilized, what services will occupy it for how long and in what amounts.
- *Transport Unit (TU)*: (the information layer counterpart of) the units of goods for which services are requested and that are handled by resources.
- *Operation*: atomic action carried out by a single resource on a single TU starting at a certain time and continuing for a certain time duration, consuming a certain amount of resource capacity.

Figure 1 Object model



Object model

The SCAL system is based on the (object) model shown in Figure 1 (arrows denote the most important relationships). In this figure, the services side is representative for the information network, the resources side represents the logistics network. In the next sections, the interpretation of this model will be further clarified.

Service requesting

A client creates a service (as an entity), and defines service requirements (in terms of place and time, and special treatment), which are stored in the entity. This service starts making a plan, a path through the logistics network in accordance with the service requirements, in fact being a list of operations ordered in time together constituting the service. It does so by creating for that purpose a planner (or using an existing one), the "service agent"; this planner is the most active part at the services side of the system.

The planner makes a plan by consulting a world model of the logistics network, which is available globally or which the planner maintains itself. (This world model is not shown in the above picture.) It tries to find an optimal path through that network by entering a step by step algorithm or by selecting some alternative possibilities, evaluating them and choosing the best one. The solution is most probably time dependent. Especially, if transports are time table scheduled, throughput time will be dependent on the exact moment the service request enters the system. Finding an optimal path or evaluating alternatives necessitates visiting relevant resources and negotiating possible solutions.

After execution has started, the planner follows the progress of the service being carried out. As soon as real world circumstances cause deviations from the set plan, the planner tries to adapt the plan. It will look for the part not yet realized and makes a new plan if possible. In this way, the plan is as actual as possible.

Resource providing

The resources side of logistics control systems, at least their structure, is in general far more static than the services side. Making investments in resources is a long term business. Logistics operators decide to invest in a certain kind of logistics equipment, functioning as a resource. This may include sorting centres, transports, docks (for loading

Logistics Information Management
Volume 14 · Number 3 · 2001 · 185–195

and unloading), internal transports, warehouses, etc.

A resource needs to know what to do at any moment, what service is the subject of its operations at what time and with what requirements. It therefore maintains a schedule: a list of operations ordered in time together determining the resource's workload. It does so by creating a scheduler.

It is this scheduler that welcomes service planners to negotiate possibilities and conditions (e.g. costs and time) of the underlying resource. The scheduler makes reservations and, on confirmation from the service planners, builds the schedule for its resource. It also removes operations from the schedule if plans have to be changed, and initiates rescheduling if necessary.

TUs and operations

Damen (1994) has introduced the concept of TU, functioning as an abstract of all the goods a logistics system has to process. This regards not only the basic goods for which services are requested, originating from outside the system, but also all units of transport made within and by the system itself, like trays, bundles, bags, containers, pallets, for short, transport means of all kinds. In our prototype system, described later, parcels are the main type of TU. It can be disputed whether some TUs are resources or not, especially all types of autonomous transports. We treat them all as TUs and we interpret the transports regarded as activities carried out with them as resources. Trucks are the only type of (road) transports used in our prototype.

There is a direct link between the concept of the TU and the information layer entity describing the goods for which services must be carried out. Both refer to the passive entities in the system that undergo all kinds of actions. These actions are called operations. For information purposes, a service may be viewed as the entity defining the requirements, the TU is regarded as the description of the physical goods and the operation is the definition of any of the activities to be carried out to reach the requirements. One operation refers to one TU and one resource, and specifies the time (moment and time window) the action is planned (and scheduled) to take place.

Operations link services to resources. They are the cutting edge of the conflict between services and resources, referred to above. Their differing "dimensions" may be depicted in three-dimensional space, like in the next picture (see Figure 2).

In this picture, the x-axis represents services, the y-axis denotes resources, and time runs up vertically, along the z-axis. Operations are depicted as vertical bars anywhere in three-dimensional space: during a certain time, some resource carries out an operation for some service. A service lies within an x-z plane, while a resource stays in a y-z plane.

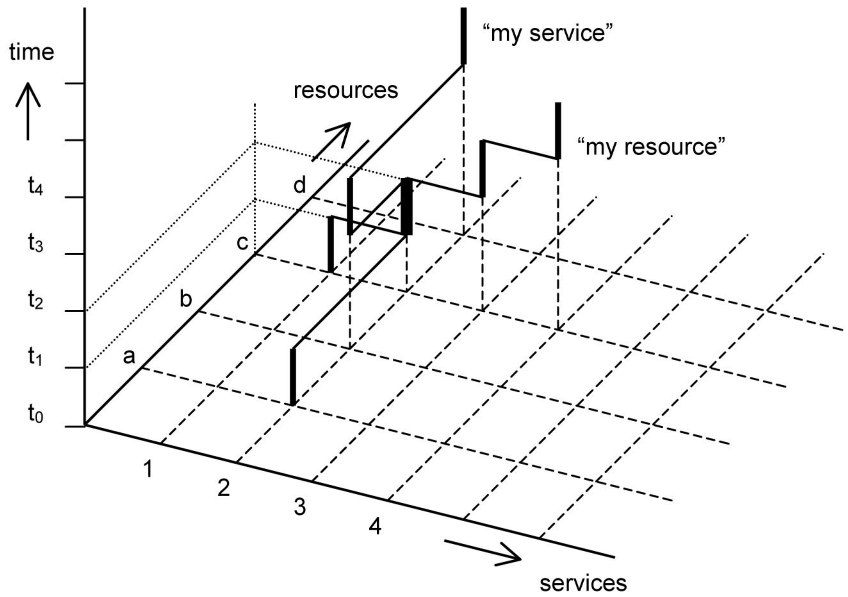
Here, only one service and one resource are displayed. From t_1 to t_2, resource c ("my resource") is busy with service 2 ('my service"). Within a certain service, operations go from one resource to another. Between succeeding operations, no time is left, denoted by a horizontal line: all time used must be represented by operations. The sequence (i.e. the plan) must guarantee, that links between succeeding resources are geographically correct. In real life, thousands of resources and many tens of thousands of services all compete for operations; the three-dimensional space is heavily occupied by vertical bars. Planning can be seen as finding consistent sequences of operations through this "forest of bars".

World model

Since service planners have to plan consistent paths through the logistics network, they have to know how the world of resources looks; they need information about the logistics network. They must know beforehand where resources are situated geographically, and roughly what kind of operations can be expected from them, in order to be able to select relevant resources. Therefore, a world model in one form or another is needed which must contain at least a list of all resources accessible by service planners, together with their main characteristics. Generally, the network will consist of network nodes and many kinds of transportation between these nodes. Indoors activities are carried out at these nodes like cross-docking, sorting, order picking, storing and retrieving, etcetera.

The world model can be realized in many different ways. Although it is possible, a centralized approach seems not very obvious taken the distributed character of the SCAL system. A hierarchical approach looks more natural in which most service planners have a

Figure 2 Services and resources as different dimensions



rough knowledge about a large part of the world (that they maintain themselves or they get from specialized centres), while less and less planners know more and more about smaller parts in which they are particularly interested.

Planning and scheduling

Planning is the activity of making a plan for a new service request. It can also consist of updating an existing, possibly partly realized plan of a service. This will happen, when real world circumstances lead to deviations from the original plan.

A plan consists of a complete, time-ordered list of operations that must be carried out in order to realize the service. Various strategies for deriving a plan are possible:

- *Parallel approach*. The planner first selects several alternatives that seem to solve the service requirements on a not too detailed level. It then evaluates each alternative by visiting the relevant resources (schedulers) and negotiates about conditions (time and costs). Finally, it chooses the best alternative it can find.
- *Path finding approach*. The planner uses algorithms to find an optimal path through the logistics network step by step (e.g. Floyd's algorithm, Dijkstra's algorithm). It therefore visits all relevant resources (schedulers) during each step to find out which one will be chosen for that step. It goes on until the destination is reached; it then has found the best solution.

The negotiation between planners and schedulers boils down to the following steps. The planner gives the desired start time for a particular operation the resource should carry out for it to the scheduler. The scheduler answers with the best fit of that operation within its current schedule at that particular moment. If it is acceptable to the planner, it reserves the operation. Later on, when a decision about the final path has been made, the planner confirms the reserved operation if chosen, or has it deleted by the scheduler if not used. Every time a plan must be updated, the same negotiations take place for the part of the plan to be changed. Of course, deleting confirmed operations is different from deleting reserved operations, especially in costs! However, ready made plans are generally updated only when resources do not actually comply with the confirmed (and agreed upon) operations, mainly due to real world circumstances.

In principle, the scheduler is totally free in how to arrange operations for its resource. The simplest way is to maintain a list of operations on a linear time scale. Here only sequence and duration of each operation need to be taken into account. This is often influenced by timetables; transports, especially, must be waited for until they leave.

This means that a lot of operations are "synchronized", since their sequences meet at the same transport. When several operations can take place in parallel (e.g. loading or unloading trucks with more people), two parameters determine the schedule: throughput time and throughput capacity. Besides the duration (throughput time: start to end time), also the number of operations per hour or some unit of time must be dealt with. Yet a third parameter is involved when capacity is determined by volume, e.g. total volume or weight capacity for truck or plane.

Looking at operations, it can be concluded that each one should be referred to twice: once from within a plan and once from within a schedule. Here, consistency is important: any change from one side must be communicated, and agreed upon, with the other side.

Prototype

A prototype of a SCAL system has been built, and a simulation of a European wide road network for express logistics for parcels has been carried out. The prototype has been dubbed INImini – a mini simulation system for integrated logistics service providing and information technology (I&I). It points to using modern information and communication techniques to boost the overall quality of logistics services realized by several, more or less independent, but cooperating, service providers in a logistics network.

Both simulation and prototype have been realized in the object-oriented language Smalltalk (VisualWorks® by ObjectShare® Inc.). It represents the information layer of INImini. A real world environment has been simulated and animated independently using AutoMod® (by AutoSimulations™ Inc.), representing the (physical) logistics network. Between the two programs, communication is carried out by simple file transfer.

INImini simulates a system for express services for parcels, with as-soon-as-possible delivery requirements. Although due end times for each parcel are parameterized, they are not used. INImini consists of three main parts:

(1) *Core*. "Core" contains the basics of the simulation engine and also contains the configuration (world model). The main constituent parts are given in Table I.

Table I Main constituent parts of "core"

Class	Major methods
Basics	
INI	go, loop, newClient, activateWorldFor:, handleDeliveredParcel:, batch
TimeClock	Clocktime, Starttime, next
RanGen	next
Dimension	length, width, height
Configuration	
Network	OperatorList, ResourceList, NodeList, initializeConfiguration
Node	neighbourResourceList, neighbourNodeList

(2) *User interface*. The user interface is aimed at making visible the various parts of the system only for development purposes and is not meant to function as an end-user interface.

(3) *Works*. "Works" constitutes the main and most important part of the package. It has three subdivisions: services, resources and operations (see Table II).

Global working

At the start of the program, a configuration file is read and the world model is set up, consisting of several tables: operators, resources, nodes. The logistics network is represented by an "activity-on-arc" graph, each arc representing one resource. The necessary list of nodes is generated by the program. Activating the "go" button, or giving the batch command, starts the simulation.

Time synchronisation with the "outside world" (the AutoMod program, or faked if off-line) is important and is established by a hand-shake via two files. After "go", a background loop is started from which basically only one activity is carried out: reading the "input" file and, from that, deciding what to do next. There are two possibilities:

(1) Time to generate a new service request: a newClient method is activated.

(2) Some truck has arrived. This is the place for special events from outside to trigger special actions, but, currently, truck arrival is the only special event reacted upon, all other operations are taken to be carried out according to the plan.

When a new service request is generated, this results in a chain of actions. First, newClient sends requestService to a new client. A new service is made with new TU, start time and

Table II The three subdivisions of "works"

Class	Major methods
Services	
Client	serviceList, addService:, releaseService:, requestService, serviceDelivered:on:at:
Service	ownerClient, serviceTU, sender, receiver, startTimestamp, endTimestamp, servicePlan, rePlanFrom:
ServicePlan	ownerService, operationList, planValue, planner, addOperation:, changePlanFrom:
ServicePlanner	ownerPlan, makePlanFor:, makeNewPlanFrom:, cancelPlanFor:from:
Resources	
LogisticsOperator	OperatorNameAndAddress, resourceList
Resource	ownerOperator, resourceSchedule, maxTUs, maxCapacity, resourceCost, resourceState, timeDuration, timeTable, kindOfOperation
ResourceSchedule	ownerResource, operationList, resourceScheduler, addOperation:, releaseOperation:
ResourceScheduler	ownerSchedule, reserveOperationFor:, confirmOperation:, deleteOperation:, cancelConfirmedOperation:
Operations	
Operation	operationResource, operationTU, startTimestamp, endTimestamp, operationState
SortOperation	nextDestination
TransportOperation	truckTU, arrived
TransportUnit	barcodeID, description, dimension, weight, ownerService, containedTUs, numberOfTUs

end time. Upon making a new TU, a sender and receiver are statistically drawn, possibly according to a given distribution. The new service instantiates a new servicePlan, which in turn creates a new servicePlanner.

The servicePlanner negotiates with resourceSchedulers (which are created during setup of the world model) and creates an as good as possible plan for the service, given the current situation. Upon confirmation of the operations, each resourceScheduler sends the to do operations to the simulation engine, that puts them into the eventlist, the backbone of the simulation engine. From there, the operations are put into a command file and sent to the outside world to be carried out.

If on truck arrival the truck is too late, the current plans for all the parcels in the truck are changed: the part of each plan not yet realized is deleted and exchanged with a new (partial) plan, starting at the arrival point.

Some results

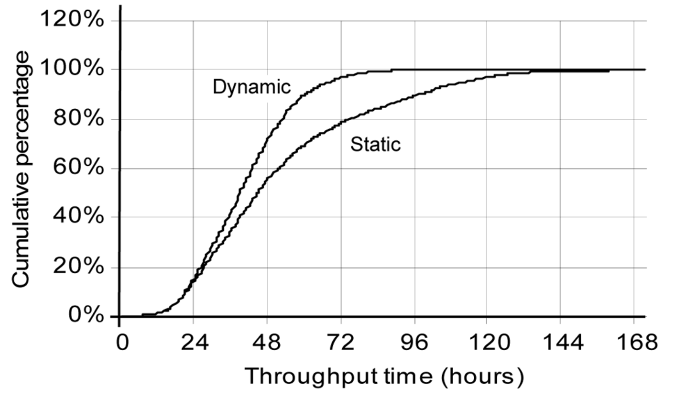
In distribution logistics, the standard approach to deliver express services is to make a choice out of a limited set of predefined services, mainly differentiating in time-cost combinations (e.g. TNT, 1998). Then a "default" path through the network is chosen, set beforehand for each possible choice.
The flexibility and dynamic behaviour of the INImini approach is twofold:

(1) It looks for possible alternative paths through the network for a given sender receiver combination, and chooses the best possibility for that particular moment. This depends mainly on truck timetables. When, for instance, a truck for a certain path has left recently, choosing a different path probably gives a better result.
(2) It reacts on deviations from the planned service caused by real world changing circumstances. Within INImini, this happens when trucks arrive too late. For each parcel included in such a truck, replanning is effectuated: the remainders of their plans are changed (if necessary) and adapted accordingly.

Simulations have been carried out for both "old" and new cases, i.e. the standard approach with default routes as well as the approach with choices from alternative routes. Typical results are depicted in Figure 3.

The upper curve gives the results of the new, dynamic planning approach, the lower curve for the conventional, default approach. Each curve represents a cumulative histogram of the throughput times of parcels: it shows the percentage of the total amount of parcels with throughput times less than or equal to the corresponding value on the *x*-axis. (For instance, from the diagram of the dynamic approach it can be seen, that 80 per cent of all parcels reach their goal in about 50 hours or less.) The results represent a simulation of seven days of an imitated European-wide parcel express service. Only parcels processed

Figure 3 Preliminary results



within the stationary period of the simulation are taken into account.

Since the various routes differ greatly in throughput times, the precise form of the curves do not carry much information. But both simulation types were carried out upon exactly the same set of parcels: start time, due end time (although not used), sender and receiver were all the same. Therefore, the curves can be compared reliably. The difference is striking: the largest difference is up to about three days, on throughput times of about three-and-a-half days (for the dynamic approach)!

Other observations on these particular simulation runs showed, that the load on the total system was so heavy, that the conventional way of processing could not keep pace with it: the total number of parcels within the system at any moment increased steadily. The dynamic planning approach on the other hand was able to process this same amount of parcels. This clearly demonstrates the capability of the new approach to better utilize the total capacity present within the system.

Future development

The prototype so far takes the configuration, representing the real world, as is. Furthermore, it is carried out by a single computer (as far as the Smalltalk program is concerned). Several future developments are possible.

The physical network is distributed by nature: resources with their schedulers occupy different places within the network. A future SCAL system will be distributed in two ways:

(1) according to the geographical distribution of the resources; and
(2) according to the placement, independent thereof, of service input points.

Service planners will have their homes at these service input points. They will travel through the information network – most likely the Internet – from resource to resource in order to negotiate and to gather information about the best possible way to implement each service. They will behave like mobile agents. On execution of the service, they follow it to check the way it is realized. If necessary, replanning is initiated, not only due to deviations from the original plans, but whenever the planner finds a better way for the service to be realized. This makes planning a completely dynamic activity.

By nature, resources behave in a more static way: the fixed geographical position within the physical network, set capacities, timetables, etc. But they are not at all confined to that. The symmetry of the model intuitively leads to a more dynamic behaviour by resources as well.

A future SCAL system could look like as follows. Resources are distributed according to the physical network; their local control systems are connected to the Internet. Service input stations are situated where convenient, also connected to the Internet. Resources advertise their possibilities on the Internet, not only static information but especially time dependent capacity availability. Service planners search on the Internet (most probably by subscription) for information to build a world view and keep it up to date. To make the symmetry complete: planners should "publish" their interests on the Internet too. Consequently, a totally dynamic interaction system could develop, in which resources look for whoever could be interested in their offerings, and service planners look for all kinds of possibilities to realize services the best way they can find.

So far, the model is used for a logistics control system. It is, however, appropriate for any system obeying the following wide interpretation of logistics control: realizing services by more or less independent, geographically distributed resources.

Object orientation gives strong support for a further abstraction of the model. The generic model created this way could serve as a blueprint for dynamically controlling such systems. It seems particularly appropriate for any kind of workflow management, above all those dealing with a combination of physical goods like forms, documents and pictures, and data processing like conversion, storage and retrieval.

From the details of the prototype, it can be concluded that a sharper distinction between TUs and resources may be necessary. Especially in a more generic approach, the TU represents a *workload*, whereas resources represent *workcapacities*.

From these, we have compiled a "wish list" of recommendations:

- Extend the prototype to a fully distributed control system, in which logistics resources are distributed according to their geographical characteristics, and in which the service planners act as real mobile agents making use of the Internet to visit the schedulers at the logistics resource's local control systems to negotiate about operations.
- Make the resources side as dynamic as the services side of the system, by enabling logistics operators to insert or remove resources at any time, or to change capacity of existing resources dynamically.
- Test several (new) dynamic planning strategies in order to find the best possible solutions under varying conditions. Particularly, the interrelationships and mutual influences of service planners should be studied in more detail.
- Make the model more generic. It then may be applicable as a blueprint for other areas, like workflow management. In addition, search for generic, basic functions for the control layer (analogous to the basic functions in Damen, 1994), and build these into the generic model.

Conclusions

Current distribution logistics systems lack the responsiveness upon rapidly changing real world circumstances, mainly due to being too much focused on the physical processing of the goods.

Shifting overall control away from the physical processing layer towards an information and control layer opens up opportunities for being more client (and service) friendly, as well as being more responsive to changes from outside.

The presented SCAL system realizes this shift. It is based on a new model of logistics, in which a clear distinction is made between service requesting, representing the control layer and resource providing for the physical processing of goods.

This model makes use of the (geographically) distributed nature of logistics systems and the principal independence of goods processing subsystems.

It solves the problem of lack of responsiveness as well as guaranteeing the overall quality of services, carried out by distributed, independent parties, called integral logistics service providing.

It does so by making the planning of services the exclusive responsibility of the services themselves, by introducing service planners that act like mobile agents negotiating with available resources about necessary logistics operations.

In a prototype system, the working of the model is tested by simulating express logistics, using a European road network as an example. Simulations show that:

- it is possible to realize logistics control systems in which the services themselves act as autonomous entities, controlling the overall realization of the service requirements and leaving the burden of handling goods to distributed, independent physical processing subsystems;
- dynamic planning can be enabled by service planners, that follow the service execution and (before or during execution) change set plans if real world circumstances necessitate or "invite" to do so; and
- considerable improvements are possible, especially if logistics resources are heavily loaded.

References

Babeliowsky, M.N.F. (1997), "Designing interorganizational logistic networks: a simulation based interdisciplinary approach", dissertation, Delft University of Technology, Delft.

CrossFlow Project, (1998), *Cross-Organisational Workflow Management*, Esprit Project E/28635, available at: www.crossflow.org

Damen, J.T.W. (1994), "Flexible mail processing: an application of integral postal engineering", PhD thesis, Technical University Delft, Delft, April.

Evers, J.J.M. (1998), "Agile logistics system: the case of a container terminal", paper presented at the GLORI Conference, Warwick, September.

Evers, J.J.M., Loeve, L. and Lindeijer, D.G. (2000), "The service-oriented agile logistic control and engineering system: SERVICES", *Logistics Information Management*, Vol. 13 No. 2, pp. 77-90.

GLORI (1999), Global Logistics Research Initiative, available at: www.glori.com

Greis, N.P. and Kasarda, J.D. (1997), "Enterprise logistics in the information era", *California Management Review*, Spring, pp. 55-78.

Gunasekaran, A. (1998), "Agile manufacturing: enablers and an implementation framework", *International Journal of Production Research*, Vol. 36 No. 5.

Hoogeweegen, M.R. (1997), "Modular network design: assessing the impact of EDI", dissertation, Rotterdam School of Management, Erasmus University Rotterdam, Rotterdam.

IDEAS (1994), Information-controlled Distribution with Enhanced Automation Systems, BriteEuram project 7494, European Commission, April.

Kidd, P.T. (1994), *Agile Manufacturing: Forging New Frontiers*, Addison-Wesley, Reading, MA.

de Kok, A.G. (1998), "Logistiek is niet intelligent genoeg (Logistics is not intelligent enough), Gericht investeren in kennis van processen en beslissingsondersteuning", *LogistiekKrant*, No. 10, 4 June (in Dutch).

Narasimhan, R. and Das, A. (1999), "Manufacturing agility and supply chain management practices", *Production and Inventory Management Journal, American Production and Inventory Control Society (APICS)*, Vol. 40 No. 1. pp. 4-10.

Sheombar, H.S. (1995), "Understanding logistics coordination: a foundation for using EDI in operational (re)design of dyadical value-adding partnerships", dissertation, Katholieke Universiteit Brabant, Brabant.

Taylor, G.D. (1995), "Agile manufacturing in the electronics industry", *NEPCON East '95, Proceedings of the Technical Program*, Boston, MA.

TNT (1998), *Nationale & Internationale Express-Services* (Brochure, in Dutch).

Verwijmeren, M.A.A.P. (1998), "Networked inventory management by distributed object technology", PhD thesis, Eindhoven University of Technology, Eindhoven.