Incomplete DRAFT

Infrastructure Software Systems

Dines Bjørner, UNU/IIST, Macau

1996

The aim of this mostly analytical paper is to propose a definition of *what an infrastructure is*. This definition will be stated in terms of properties of properties of mathematical models of the domain of the 'infrastructure', in terms of properties of requirements put to computing & communications support, and in terms of characteristics of possible architectures for software that support the 'infrastructure'.
The paper is presently in draft form, but is circulated in order to solicit comments.
Eventually a version of the paper will be presented at the Spring Session of the Mathematics and Informatics Section of Academia Europaea, 11–12 April 1996 in Amsterdam, The Netherlands.

Contents

1 A Railway System: an Example Infrastructure

In this section we develop an example of an infrastructure.

The section unfolds in several stages. First we narrate the manifest objects: components and the functions and operations on these components and the behaviours over them. Then we present a formal description. We claim that the description models the narrative. On the basis of the model we can extract a theory: a set of properties that hold true of the model. A variety of practical uses of the theory (or the model) are covered in a next subsection and a formalisation presented of one of these uses. The two formal models are then related. The section ends with a discussion of the infrastructure concept so far investigated.

The example is that of a railway system. We now list various facets of a railway system. The list is not exhaustive — that we leave to the interested reader! The purpose of showing the list is to indicate the breadth and depth of our concern. The concern is that of drawing as large a boundary — throwing as large a net — around a great variety of "things"[1] that all have to do with *being an infrastructure*, in particular: *being a railway.* We are not interested in a particular "corner", such as for example a 'passenger ticket reservation sub-system'. We are instead interested in covering "all" such infrastructure sub-systems. Only then do we believe we get close enough to capture *what is an infrastructure.* Our interest is in uncovering "all" interfaces between these 'infra-sub-structures'. We wish to investigate their sharing of possibly dynamically varying information, the flow of information between the sub-systems, and the 'invocation' of actions "across" interfaces. We perform the same experimental analysis in many subsections of section 3. Only then can we expect to get close enough to understand the infrastructure concept. This paper is a first such comparative attempt.

1.1 A Narrative: Syntax & Semantics

We group our (albeit superficial) analysis of what it means to be a railway into clusters of components and actions:

1. The Rail Network:

[1] Double quoting certain words is a *tongue-in-cheek* writing style that is used to indicate that the author does not quite believe that the usage of the word is scientifically sound.

A "Top-down" Description: The rail network consists of lines and stations. A line is a sequence of blocks. A block is a sequence of linear rail units. A station can be abstracted as a set of tracks. Tracks are sequences of linear rail units. Any line connect two stations. Any track is connectable to one or more lines. Lines may be either UP, DOWN or BOTH lines with respect to traffic direction (where UP (etc.) is an arbitrary notion!). Any station is connected by one or more lines to some other stations. Stations have unique names[2]. Lines and tracks usually have attributes such as length, etc.[3]

Lines and tracks are laid, put into service, operated, checked (for wear and tear) and taken out of service, including maintained and repaired.

A "Bottom-up" Description: Another way of presenting the above is to start from basic components: Connectors are simple identifiers and are part of units. The rail network is composed from units. Units are either linear, in which case they have two distinct connectors to which other units can be attached, or are switches, in which case they have three distinct connectors, or units are simple cross-overs (four distinct connectors), or units are switch cross-overs (again four distinct connectors), etc. No three or more units can have the same connectors. Two units can share at most one connector. Two units sharing a connector (identifier) means that these two units are connected at that connector. A block is a sequence of linear units such that neighbouring units share connectors. A line is a sequence of blocks such that neighbouring units of two blocks share connectors. A track is a sequence of linear units (as for a block). A station is a set of units. A station contains one or more tracks. Lines connect stations. A line connecting to a station shares a connector with one unit of the station.

Etcetera — as per the first, top-down, Railway Network description.

2. The Rolling Stock:

   The rolling stock consists of rolling units: locomotives, possibly tenders, passenger cars, freight wagons, and special service cars. One view of a train is that it is a linear composition of one or more rolling units.[4]

   Acquisition of rolling stock may be planned and rolling stock units acquired, put into service, operated, checked (for wear and tear) taken out of service, including maintained and repaired, and dismantled. Other operations are possible.

3. The Timetable(s):

   A timetable describes for every train number and day in the year, a sequence of station visits. Each station visit is characterized by the name of the station visited, the times of arrival and departure, and, usually, the number of the platform-track or through-track at which the train is expected to stop, respectively pass through.[5] We may introduce

[2] Note that we have 'abstracted away' the notions of point machines (turnouts, switches), signals, level crossings, cabin posts, etc. Also we have not introduced such notions as shunting and marshaling yards, garages, etc.

[3] Here we 'abstract away' the topological layout of rail units and their composition. That is we 'abstract away' curvatures, gradients, whether a line or a track passes over a bridge or through a tunnel, etc.

[4] We may impose a notion of simple and composite rolling entities, where a simple entity is a rolling unit and a composite one is a sequence of one or more units.

[5] Thus we 'abstract away' timetable indications of train service facilities (whether, and if so, when, there is a restaurant car, etc.), 'corresponding train' connectabilities, etc.

several timetable notions — but refrain from doing so at present. Timetables may include passenger and freight journey price (i.e. cost) information.

Timetables may be constructed (subject to a possibly great variety and mass of constraints), put into effect or decommissioned; entries may be modified, including deleted, and new entries inserted; etc.

4. The Train Traffic — The Dynamics of the Network:

   The dynamics of train movement, signaling (switch and signal changes), passenger information (say on platform monitors), etc. is an integral part of being a railway. So by train traffic we understand the instantaneous position of trains on the network and the state of units (open (including directions of being 'open'), closed, and switch setting)[6] at any one (continuous!) time.

   Units may be opened or closed for traffic in specified directions — possibly through the setting of signals and switches. The settings may involve entire tracks or lines, or just blocks within lines. Trains may be dispatched, and their journey monitored and otherwise controlled; etc.

5. Schedules:

   A schedule is a plan for the position of all timetabled trains and the state of all rail units for a given time interval, often from some time in the not too distant past to some time in the not too distant future.

   Given a network, the rolling stock, a timetable, and possibly a great variety and mass of constraints, a schedule may be constructed; a schedule may be changed (re-scheduling); etc.

   Train traffic may be monitored with respect to adherence to schedule.

6. Staff:

   Trains, stations, etc. are operated by railway staff. Some staff manage the setting of the states of units according to schedules. Others accommodate bookings, advise passengers on boarding and alighting from trains, deal with the loading and unloading of freight, dispatch trains, or drive trains. Etc.[7]

   Staff are allocated and scheduled; staff operate electronic, electro-mechanical and mechanical gear; staff plan, supervise, monitor and otherwise control resources (including disbursement of funds); and staff otherwise interface with rolling stock, other railway system facilities, other personnel, passengers and freight agents.

7. Bookings:

   Bookings record, for every scheduled train (or a reasonable subset of the timetabled trains), which passenger seats and which freight compartments have been reserved (including: are occupied) together with booking information.

[6] We again 'abstract away' that which determines unit and block states: typically the state of signals — which have yet to be introduced (and which will not be introduced at all!).

[7] We here cut short our treatment of staff — thus omitting crucial staff attributes necessary for the purposes of allocating and scheduling staff.

8. Passengers:

   Passengers book (modify and cancel) trips and board and alight from trains. Potential passengers inquire about timetables, the state of the traffic, the arrival of freight, etc. Passengers occupy seats, or fail to honour a reservation.

9. Freight:

   Freight arrives at, is stored at, and leaves stations for loading, awaiting pick-up, and after delivery respectively. Freight occupies wagon space.

10. *&c.:*

    We end our presentation of components and operations over them.

1.2 A Proposed 'Base' Model

We only present the signature of the model, and we only show some parts of the signature, hopefully enough to justify later claims. For a full model we refer to [2]. We make no reference to staff, and passengers and freight are described only very cursorily.

The model follows the bottom-up description given above.

type

C, U, Tn, S, T, L, P, N, M, Q, W
$\Sigma = (\mathrm{C}\times\mathrm{C})\textbf{-set}$
$\Omega = \Sigma\textbf{-set}$
$\mathrm{TT} = \mathrm{Tn} \overrightarrow{m} (\mathrm{S} \overrightarrow{m} (\mathrm{T} \times \mathrm{T}))$
$\mathrm{RE} = \mathrm{Tn} \overrightarrow{m} ((\mathrm{S} \times \mathrm{S}) \overrightarrow{m} (\mathrm{M} \overrightarrow{m} \mathrm{Q}\textbf{-set}))$
$\mathrm{TF} = \mathrm{T} \xrightarrow{\sim} (\mathrm{Tn} \overrightarrow{m} \mathrm{U}^*)$
$\mathrm{RS} = \mathrm{W} \overrightarrow{m} \mathrm{U}^*$

value

inv_L: L → **Bool**
inv_S → **Bool**
inv_N → **Bool**
inv_TT: TT → **Bool**
inv_RE: RE → **Bool**
inv_TF: TF → **Bool**
inv_RS: RS → **Bool**

inv_TT_N: TT × N → **Bool**
inv_TT_RE: TT × RE → **Bool**
inv_TF_N: TF × N → **Bool**
inv_TF_TT_N: TF × TT × N → **Bool**
inv_RS_N: RS × N → **Bool**
inv_RS_(TF_T)_N: RS × RS × (TF × T) × N → **Bool**

value

units: U: **U-set**
l_units: U: **U-set**

```
    r_units: U: U-set

axiom
    $\forall$ u:U $\bullet$
        l_units(u) $\cap$ r_units(u) = {} $\wedge$
        units(u) = l_units(u) $\cup$ r_units(u)
        let cs = units(u) in
            $\forall$ c:C $\bullet$ c $\in$ cs $\Rightarrow$
                card { u$'$ | u$'$:U $\bullet$ u$'$ $\neq$ u $\wedge$ c $\in$ units(u$'$)} $\leq$ 2
        end

value
    state: U $\to$ $\Sigma$
    states: U $\to$ $\Omega$
    xLUs: L $\to$ U-set
    xPUs: P $\to$ U-set
    xNLs: N $\to$ L-set
    xNSs: N $\to$ S-set
    xSPs: S $\to$ P-set
    xinLsP: S $\times$ P $\overset{\sim}{\to}$ L-set
    xoutLsP: S $\times$ P $\overset{\sim}{\to}$ L-set
    xin: S $\times$ L $\overset{\sim}{\to}$ P-set
    xout: S $\times$ L $\overset{\sim}{\to}$ P-set

value
    Open_Route: U $\times$ U $\times$ N $\overset{\sim}{\to}$ N
    Close_Route: U $\times$ U $\times$ N $\overset{\sim}{\to}$ N
    ...
    Sched: TT $\times$ N $\overset{\sim}{\to}$ TF-set
    BestSched: TF-set $\times$ N $\times$ Criteria $\overset{\sim}{\to}$ TF
    Disrupt: TF $\times$ TF $\overset{\sim}{\to}$ T
    ...

type
    Criteria = (TF $\times$ TF) $\to$ Rat
```

Annotations:

Components:

1. C, U, Tn, S, T, B, L, P, N, M, Q, W: These sorts stands for C:Connectors, U:rail Units, Tn:Train numbers, S:Stations, T:Time, B:Blocks, L:Lines, P:Tracks (including platforms), N:Networks, M:Clients, Q:reservation units (passenger seats or freight space locations), and W:rolling stock units.

2. Σ, Ω: These type names models the states, respectively the set of states, that a rail unit is in, respectively can range over. The type $C \times C$ models the set of paths through a

rail unit.

3. TT: Timetables maps Train numbers to station visits, with station visits mapping stations to train arrival and departure times.

   (For trains originating at a station the arrival time designates the time at which the train is first made ready at the platform. For trains terminating at a station the departure time designates the time at which the train is taken out of service at that station's (arriving) platform.)

4. RE: Reservations map, for each train number, and for each pair of stations (on the train's journey, i.e. along the sequence of station visits), and for each client, that client's reservation (and/or actual occupancy) of seat or freight space unit locations.

5. TF: For each real time and for each train number the traffic identifies the sequence of units (to be) occupied by that train at that time.

6. RS: For each unit of non-train rolling stock (passenger car, freight car, locomotive, tender, etc.) rs:RS shows its location.

   <u>Invariants:</u>

7. inv_L: A set of units is said to form a line if they are all linear units and connectable in a sequence.

8. inv_S: A set of units is said to form a station if it contains at least one track, etc.

9. inv_N: A set of units is said to form a network if it contains at least two stations and if all stations are connected to some other station by a line, etc.

10. inv_TT: Departure time at a station must always be larger, by some station-dependent amount than the corresponding arrival time. The ordering of time implies an ordering on station visits, so that arrival time at a station must be later than departure times from preceding stations.

11. inv_TF: At no point in time can two or more trains occupy overlapping units. At any pair of infinitisimally separated times a train has at most made a simple move: entering a unit, leaving a unit, or both (— or made no unit changes). For any two times for which a train is recorded it must be the case that the same train is recorded in the traffic at all intermediate times (there can be no ghost trains). Etc.

12. inv_RE: No two distinct clients can occupy overlapping passenger or freight space units.

13. inv_RS: No two rolling stock units can occupy overlapping rail units.

14. inv_TT_N: The stations mentioned in a timetable must be stations of the network, and a train journey (the sequence of stations visited) must be stations along some route of the network.

15. inv_TT_RE: A pair of stations between which a reservation has been made on some train must lie on the route of that train in the timetable.

16. inv_TF_N: Traffic must be commensurate with the network: lists of units mentioned in the traffic must correspond to potentially open routes of the network; a train journey starts and ends at distinct stations; etc.

17. inv_TF_TT_N: Traffic must be on time and otherwise according to schedule. (This constraint may be violated.)

18. inv_RS_(TF_T)_N: Rolling stock can at the time specified not interfere with the traffic: no rolling stock unit can occupy units of trains en route; etc.

    Observers:

19. units, l_units, r_units: Each unit (U) is characterised by its distinct connectors (C).

20. **axioms:** state the above and that at most two units can share connectors, and from the afore, then exactly one connector. The axiom thus amounts to the **syntax** rule for composing units into networks of lines and stations.

21. state: Given a unit (at any time, not specified) this function observes its (current) state.

22. states: Given a unit this function observes all the states it may range over.

23. xLUs: Given a line this function observes the set of linear units that it consists of.

24. xPUs: Given a track this function observes the set of linear units that it consists of.

25. xNLs: Given a network this function observes the set of lines of the network.

26. xNSs: Given a network this function observes the set of stations of the network.

27. xSPs: Given a station this function observes the set of tracks (platforms, etc.) of that station.

28. xinLsP: Given a station and a track this function observes the set of lines which are connected to that station and from which traffic can arrive to this track.

    (Observe that we postulate that the station has enough knowledge about the lines in question. That is: we postulate that there is not necessarily a need to invoke the network as an argument.)

29. xoutLsP: Given a station and a track this function observes the set of lines which are connected to that station and to which traffic can depart from this track.

    (Similar observation as just made above.)

30. xin: Observes the tracks that can be reached from an input line to a station.

    (Similar observation as made above.)

31. xout: Observes the tracks in a station from which a line can be reached.

    (Similar observation as made above.)

    Many other observer functions can be postulated. Note that they may not be definable on the basis of the combined abstract (sort) and concrete type definitions given so far.

Generators:

32. Open_Route: A route is a sequence of connected units. An open route is such a sequence in which the state of each unit allows traffic in the direction of the sequence as indicated by the pair of argument units.

33. Close_Route: A closed route is a route in which the state of each unit is the empty set. That is, all paths of each unit are closed.

34. Sched: To schedule is to provide a set of feasible traffics. All that is basically needed for that is a timetable and the network of rail units. A timetable may be such that no feasible traffics are possible, or such that more than one traffic is feasible. Hence scheduling may, in general provide a set of solutions.

    (The Sched[ule] function typically represents a constraint satisfaction problem.)

35. BestSched: A best schedule can be obtained by applying a criteria function that, for example, compares any two traffics with respect to the timetable and determines which is the better.

36. Disrupt: A real traffic is said to represent a disruption with respect to a scheduled traffic at some first time if up to that time the two traffics 'coincide', but at that time they do not. 'Coincidence' may be measured in terms of one or more trains not occupying the units they were scheduled to be located at.

37. Criteria: The higher a traffic rates with respect to a timetable the better it is said to be.

    (The Criteria function typically represents a constraint satisfaction problem.)

1.3 Towards A Theory of Railways

Once we have defined the various functions (invariants, observes, generators, etc.), for example in terms of axioms, we can develop a theory. Examples of postulated theorems are:

1. Kirschoff's Law of Train Traffic:

    - Assume two distinct times such that at both of these times there is no train traffic.
    - The two times determine a period.

      Then:
    - The number of trains flowing into any station in that period,
    - (-) minus the number of trains terminating at the station in the period,
    - (+) plus the number of trains originating at the station in the period,
    - (=) equals the number of trains flowing out of the station in the period.

2. Preservation of Traffic Order:

    - Let two trains, A and B, be moving in the same direction along a line or track, and assume A is in front at time t.
    - Assume a later time t′ at which both trains are still moving along the same line (or track).

Then

- A will still be in front: the two trains cannot have changed position relative to each other!

3. Liveness:

   -
   -
   -

4. Conservation:

   -
   -
   -

5. Monotonicity:

   -
   -
   -

1.4 Pragmatics + more Syntax & Semantics

The syntax and semantics description of the railway system focused on manifested components and functions, operations and behaviours (e.g. traffic) over these. The level of focus was "gadget-oriented"[8].

In reality the management and operation of a railway system is dictated by a number of less physical but more pragmatic concerns. By a pragmatic concern we mean one that focuses on the socio-economic context in which the system is used and includes how its operation is perceived by the humans involved.

As we shall hopefully demonstrate, the pragmatic concerns (or view) are in harmony, that is: are not in conflict, with the syntactic and semantic view.

Next we present a list of pragmatic, management, operational and usage views of the railway systems. Each of the views is first given a label, then some key words, and finally some *textual* explanation.

The list is ordered as follows: items 1–3, and within item 3 sub-items 3a–3d, are in order of increasingly shorter time interval between "cyclic", repeated invocations, and hence response times.

Functions and operations of item 1 usually have a time span of up to typically five years. Functions and operations of item 2 usually have a time span of between three and 12 months. Functions and operations of item 3a usually have a time span of a month or so, of item 3b of usually between a week and just hours, of item 3c of usually five to 10 minutes, and of item 3d usually minutes down to seconds! Finally statistics functions are usually invoked weekly to monthly. Results of statistics functions are "fed back" into strategic and tactical planning.

[8] Gadgets are physical objects. So you may call the above focus: "object-oriented".

1. Strategics: Planning & Development

   Resource & Service Up/Downgrade: New Lines, New Stations, New Trains, New Services, ... Related Finance and Staff Assessments, Investments. Staff Development: Awareness, Training, Testing, ...

   > *Strategic planning and development involves the network, lines, stations, trains and journeys offered (i.e. timetable possibilities), etc. Strategic planning amount to functions that may involve simulation over the above-mentioned components. Strategic planning may conclude that new, respectively old, lines and/or stations need to be developed or closed. Strategic development creates (or "destroys") such new (old) facilities. Strategic planning also estimates resource outlays: cost, staff and material expenses, respectively recovery (savings). Strategic planning is based on (projected) financial bases and statistics concerning for example projected passenger and freight loads.*

   value

   Stra_Plan: $\mathrm{N} \times \mathrm{RS} \times \mathrm{Staff} \times \mathrm{Cost} \times \mathrm{TT} \times \mathrm{Statistics}$
   $\quad\quad \xrightarrow{\sim} \mathrm{N} \times \mathrm{RS} \times \mathrm{Staff} \times \mathrm{Cost}$

   Stra_Devt: $\mathrm{N} \times \mathrm{RS} \times \mathrm{Staff} \times \mathrm{Cost}$
   $\quad\quad \xrightarrow{\sim} \mathrm{N} \times \mathrm{RS} \times \mathrm{Staff} \times \mathrm{Cost}$

   > *Strategic development is here seen as a monotonic operation: it provides 'delta' increment/decrement network, rolling stock, staff and cost resources and results in new, monotonically changed resources.*

2. Tactics: Resource Scheduling & Allocation: S&A

   Time-tabling, Rolling Stock S&A, Staff S&A, ...

   > *Based on estimated load of passengers and freight, in various time intervals and between various pairs of stations (EstLd), one can construct (TiTbl) a set of timetables from the available resources. Based on various policy (etc.) criteria (PoCri) one can then choose (BstTT) a "best" timetable.*
   > *We have seen above that timetables imply schedules. Invocation of seasonal schedules implies the readiness, hence the allocation and scheduling, of resources. This may involve constructing (RSReAllocPl) a reallocation of rolling stock according to a plan for shunting rolling stock units into trains. Executing this resource reallocation plan (RSReAllocEx) takes time and results in a new state of rolling stock. The passing of time is recorded in a revised traffic. Etc.*

   type

   EstLd = $(\mathrm{T} \times \mathrm{T}) \underset{m}{\rightarrow} ((\mathrm{S} \times \mathrm{S}) \underset{m}{\rightarrow} (\mathrm{Q} \underset{m}{\rightarrow} \mathbf{Nat}))$
   PoCri
   ShunPl = $\mathrm{RS} \times (\mathrm{Tn} \underset{m}{\rightarrow} \mathrm{U}^*)$

value
TiTbl: EstLd $\times$ N $\times$ RS $\times$ Staff $\xrightarrow{\sim}$ TT-**set**
BstTT: TT-**set** $\xrightarrow{\sim}$ TT
RSReAllocPl: TR $\times$ RS $\times$ TT $\times$ $\xrightarrow{\sim}$ (ShunPl $\times$ TR)
RSReAllocEx: (ShunPl $\times$ TR) $\times$ RS $\times$ N $\xrightarrow{\sim}$ TR $\times$ RS

3. Operations: Resource Deployment

(a) Maintenance: Preventive, Adaptive, Corrective

Based on general characteristics of resources wrt. f.ex. wear and tear (as also reported) one can construct a schedule (PMSch) of time intervals for the preventive maintenance of all resources. Preventive maintenance (PMain) now is an operation which performs the specified maintenance and thus results in an updated set of resources: the rail net, rolling stock, staff, etc. — and also a revised preventive maintenance schedule.
Based on planned development of resources, i.e. the insertion (deployment) of new and/or the removal of old resources as per a schedule (InRmv), one can perform adaptive maintenance (AMain).
Based on reported resource breakdowns (ReRpt) one can perform corrective maintenance (CMain).

type
PMSch = (U|W|...) $\overrightarrow{m}$ (T $\times$ T)
InRmv = (U|W|...) $\overrightarrow{m}$ (T $\times$ T)
ReRpt = (U|W|...) $\overrightarrow{m}$ T

value
PMain: PMSch $\times$ N $\times$ RS $\times$ Staff $\times$ ...
$\xrightarrow{\sim}$ N $\times$ RS $\times$ Staff $\times$ ... $\times$ PMSch
AMain: InRmv $\times$ N $\times$ RS $\times$ Staff $\times$ ... $\times$ PMSch
$\xrightarrow{\sim}$ N $\times$ RS $\times$ Staff $\times$ ... $\times$ PMSch
CMain: ReRpt $\times$ N $\times$ RS $\times$ Staff $\times$ ... PMSch
$\xrightarrow{\sim}$ N $\times$ RS $\times$ Staff $\times$ ... $\times$ PMSch

(b) Clients: Inquiries, Booking, Ticketing, Freighting, *&c.*

Ticketing and freighting implies the reservation (RE) and with that goes a whole lot of queries concerning possible journeys, costs, timing, etc., for both passengers and freight. Reservations (RE) state intention of actual occupation (OC). Booking actually changes the state of reservations. Ticketing itself is here seen as the act of a person or a piece of freight (M) of actually occupying a set of reserved spaces (Q). Freighting in addition to ticketing may involve a number of other resource (Σ) deployments in addition to occupation.

ReSched: TF

type

Cost = **Nat**

Journey = Tn $\times$ (S $\times$ (T $\times$ T))*

RE = Tn $\underset{m}{\rightarrow}$ ((S $\times$ S) $\underset{m}{\rightarrow}$ (M $\underset{m}{\rightarrow}$ Q-**set**))

OC = Tn $\underset{m}{\rightarrow}$ ((S $\times$ S) $\underset{m}{\rightarrow}$ (Q $\underset{m}{\rightarrow}$ M))

Σ

value

Inq: (S $\times$ S) $\times$ TT $\overset{\sim}{\rightarrow}$ (Journey $\times$ Cost)-**set**

Book: (S $\times$ S) $\times$ TT $\times$ RE $\overset{\sim}{\rightarrow}$ RE

Tck: M $\times$ RE $\times$ OC $\overset{\sim}{\rightarrow}$ OC

Fre: M $\times$ RE $\times$ OC $\times$ Σ $\overset{\sim}{\rightarrow}$ OC $\times$ Σ

(c) Services: Information, ...

Services include those of informing the public (including passengers and freighters) of changes to the timetables and traffic, etc. Information can be given over loudspeakers, on display boards (TV Monitors), etc.

type

Info

value

Inform: TR $\times$ TR $\times$ TT $\overset{\sim}{\rightarrow}$ Info

(d) Traffic: Train Dispatching, Monitoring & Control

(In our narrative and in our base model we focused somewhat unintentionally on this aspect.)

The oftentimes "split–second" decisions related to railways take place in this area: traffic is expected to be reliable, on-time and safe. Many factors not accounted for here may cause events that change the actual traffic from the desired, scheduled traffic.

Once the actual traffic starts deviating from the scheduled traffic rescheduling has to be invoked. We have already covered scheduling. Rescheduling amounts to the introduction, for a certain next time interval, of a revised, temporary timetable and its associated revised traffic.

Here at this level we should also introduce all the necessary implements that allow the railway system to ascertain adherence to schedule, but we refrain because it would lead to an "explosion" in the descriptional coverage of this example.

value

ReSched: TF $\times$ T $\times$ TT $\times$ N $\overset{\sim}{\rightarrow}$ (TT $\times$ TF)

4. Statistics

Traffic, Usage, Wear & Tear, ...

> *The various functions relating to the gathering of statistics, which are necessary for strategic and tactical planning, form the last view of "what it means to be a railway".*
> *We shall, however, not cover this facet.*

1.5 Discussion: Pragmatics vs. Semantics

In sections 1.1 and 1.4 we presented aspects of railway systems from different points of view. In section 1.1 the focus was on the manifest components, the functions and operations on them, and the behaviours over them. We say that section 1.1 and the first formal model based on that section and given in section 1.2 represent a syntactic and semantic view of manifest objects. In section 1.4 the emphasis was on the pragmatics of these manifest objects: the organisational, managerial, operational and other facets of their use by humans. That view emphasizes how the objects (components, functions, operations, behaviours) are perceived by the different human agents: staff, clients and observers.

We saw that the treatment of the pragmatics abstracted from a number of features of the base model. Thus strategic planning (item 1 page 12) 'abstracted away' the states of units, although this was not explicitly shown. We can think of each of the items of section 1.4 as representing a view. For each of these views there is most likely a "most appropriate" formal model.

In the next section we present such a formal modal for the view of train dispatching (item 3d page 14).

1.6 A Proposed 'Dispatch' Model

type

$$\begin{array}{l}
\mathrm{Ln,\ St,\ Len,\ Tr} \\
\mathrm{N = (Ln} \xrightarrow[m]{} \mathrm{Line) \times (St} \xrightarrow[m]{} \mathrm{Station)} \\
\\
\mathrm{Line = Len \times ... \times LnTyp} \\
\mathrm{LnTyp == UP\ |\ DOWN\ |\ UPDOWN} \\
\\
\mathrm{Station = (Tr} \xrightarrow[m]{} \mathrm{Track)} \\
\mathrm{Track = Len \times ilns{:}Ln\text{-}\mathbf{set} \times olns{:}Ln\text{-}\mathbf{set} \times TrTyp} \\
\mathrm{TrTyp == LINE\ |\ SIDING\ |\ PLATFORM\ |\ FREIGHT\ |\ ...} \\
\\
\mathrm{TT' = Tn} \xrightarrow[m]{} \mathrm{StVis^*} \\
\mathrm{StVis = arr{:}Time \times St \times Tr \times dep{:}Time \times Ln} \\
\\
\mathrm{TF' = Time} \xrightarrow[m]{} \mathrm{(Tn} \xrightarrow[m]{} \mathrm{((St \times Tr)\ |\ Ln))}
\end{array}$$

value

$$\mathrm{Dispatch{:}\ TT' \times N \times TF'} \xrightarrow{\sim} \mathrm{TF'}$$

Annotations

Ln, St, Len, Tr: are sorts representing lines, stations, track lengths and track names respectively.

N: A network is here concretised as the set of all its uniquely named lines and uniquely named stations.

Line: A line has a length, ... (some other characteristics), and a line type.

LnTyp: A line is either an up, a down, or a "both" line. The choice of direction (UP, etc.) is subject to certain rules. These need not concern us here. The idea is that on an UP or a DOWN line trains can only proceed in one direction, whereas on an UPDOWN line they may travel in either direction, though there cannot, of course, be two trains on an UPDOWN line travelling in opposite directions at the same time.

Station: A station consists of a number of uniquely named tracks.

Track: Each track has a length, can be reached from some (incoming) lines, can reach some (outgoing) lines, and is of a certain track type. Thus, if a train on a track originated at another station then it must have arrived on one of the incoming lines, and if that train travels on to another station it will do so along one of the outgoing lines.

TrTyp: A LINE track allows trains to pass non-stop through a station. A SIDING track is used for "storing" rolling stock ("trains"). A PLATFORM track allows trains to stop to pick up and set down passengers. A FREIGHT track is used for the loading and unloading of freight. Etc. (Some tracks may have multiple types.)

TT′: A timetable is here abstracted as a mapping from train identifiers to sequences of station visits. The sequence (called a journey) contains at least two station visits.

StVis: A station visit names the station and gives the estimated arrival and departure times, the track along which to pass, and the incoming line on which to arrive. The arrival and departure times of the first and last station visit of a journey must allow for sufficient time for passengers and/or freight to be handled. If the interval between arrival and departure times is small, say of the order of a few minutes, then the station visit designates a through station visit.

TF′: In this model traffic records, for a finite set of discrete times, the planned or observed locations of trains. These locations are either on tracks at stations or on some line between two stations.

Dispatch: Dispatch is a recurrently invoked operation. Based on a timetable, the rail network, and the current, observed traffic, Dispatch plans a new traffic.
You may think of the Dispatch operation as being invoked at the smallest time ("input time") of the argument traffic and yielding a traffic plan whose smallest time is some increment above the "input time".
The new, planned traffic implies the "dispatch" of commands to a wide variety of railway staff to ensure that the next observed traffic is commensurate with the planned traffic.

1.7 Discussion: Views and Models

The 'Base' vs. The 'Dispatch' Model: The most recently presented model focuses only on a few of the components and functions (etc.) of a railway system. It is far "smaller" than the first (base) model. What is the relationship between these two models? The answer is simple: The second model *implements* a subset of the first model. We therefore need to make precise what is meant by *implements.*

Here we take the position that 'implements' is a relation between the two models. The relation is characterized by a number of observer, projection and composition functions.

value

$\mathrm{xInfoL}: \mathrm{L} \to (\mathrm{Ln} \times \mathrm{Len} \times \mathrm{LnTyp})$

$\mathrm{xInfoP}: \mathrm{P} \to (\mathrm{Tr} \times \mathrm{Len} \times \mathrm{TrTyp})$

value

$\mathrm{projLinesN}: \mathrm{N} \stackrel{\sim}{\to} (\mathrm{Ln} \overrightarrow{_{m}} \mathrm{Line})$

$\mathrm{projStationsN}: \mathrm{N} \stackrel{\sim}{\to} (\mathrm{St} \overrightarrow{_{m}} \mathrm{Station})$

$\mathrm{projTT'TT}: \mathrm{TT} \stackrel{\sim}{\to} \mathrm{TT}'$

$\mathrm{projTF'TF}: \mathrm{TF} \stackrel{\sim}{\to} \mathrm{TF}'$

value

$\mathrm{compN'N}: \mathrm{N} \stackrel{\sim}{\to} \mathrm{N}'$

value

$\mathrm{implements}: (\mathrm{N}' \times \mathrm{TT}' \times \mathrm{TF}') \times (\mathrm{N} \times \mathrm{TT} \times \mathrm{TF}) \to \mathbf{Bool}$

$\mathrm{implements}((\mathrm{n}',\mathrm{tt}',\mathrm{tf}'),(\mathrm{n},\mathrm{tt},\mathrm{tf})) \equiv$

$\quad \mathrm{n}' = \mathrm{compN'N}(\mathrm{n})$

$\quad \wedge\ \mathrm{tt}' = \mathrm{projTT'TT}(\mathrm{tt})$

$\quad \wedge\ \mathrm{tf}' = \mathrm{projTF'TF}(\mathrm{tf})$

It is not always the case that an 'implements' relation is this simple (and in fact an equality relation).

General: In general we see that for every pragmatic view we probably have to develop (to 'implement') a special model that best captures that view, including "projecting away" all unnecessary details.

- So why construct a 'Base Model' in the first place?
- And: how "good" is the 'Base Model' anyway?

Das Ding an sich, und das Ding für Mich![9]

[9] Taking some liberty in quoting Kant!

Base Model(s): *Das Ding an sich*[10]

The 'base model' plays, or rather is intended to play, the same rôle with respect to the 'pragmatic models' as, for example, Newtonian Mechanics with respect to various solutions to problems posed in that realm.

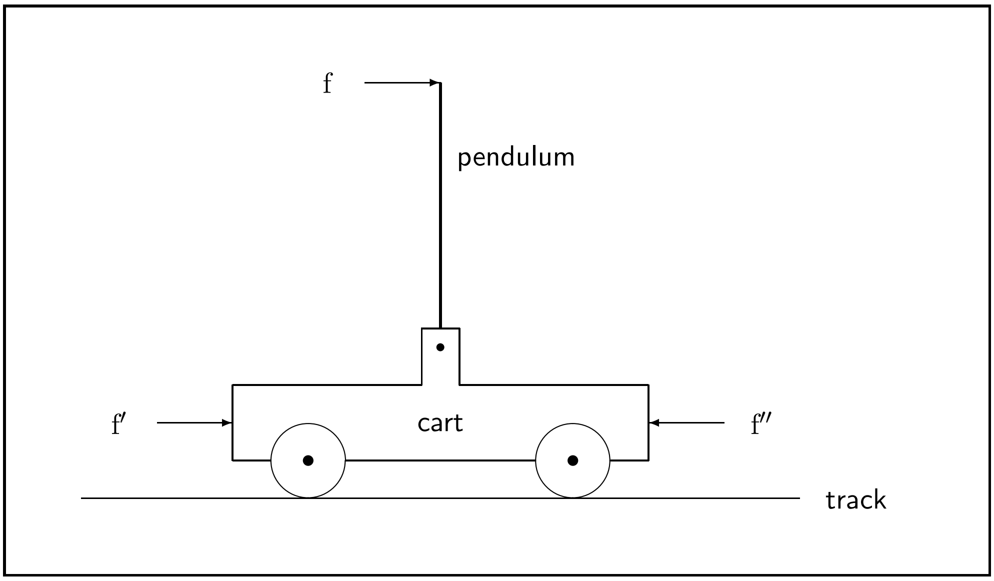
Let us explain:

Newtonian mechanics forms a base theory. It is based on the following laws:

- Newton's Law #0: The Law of Gravitation: ...[11]
- Newton's Law #1: The Principle of Inertia: *If an object is left alone, is not disturbed, with a constant velocity, if it was originally moving, or it continues to stand still if if was originally standing still.*[12]
- Newton's Law #2: *The time rate of change of momentum is proportional to the force applied.*
- Newton's Law #3: Conservation of Momentum: *Action equals Reaction.*

We appeal to Newton's laws when, for example, we wish to describe the motion (kinematics and dynamics) of an inverted pendulum on a movable cart (See Figure 1).

Figure 1: An Inverted Pendulum



The cart can only move along a straight, horizontal track. The pendulum can swing (around a revolute joint)

only in the upper half of the vertical plane through the track.

[10] Although we are not dealing with etheral things, like deep philosophical issues, but rather with man-made objects, we do take the liberty of equating "the things in themselves" with the "base model".

[11] Deduced from Kepler's 2nd and 3rd laws.

[12] Due to Galileo.

Based on Newton's laws we can write down the behaviour of this system under various conditions:

- Inverted Pendulum+Cart Behaviour #1:

  The cart is stationary and the pendulum is vertical and stable. A force f is applied to the top point of the pendulum in the plane of the pendulum movement. The mass of the pendulum is given. We assume no friction in the revolute joint.
- Inverted Pendulum+Cart Behaviour #2:

  A force f' is instantaneously applied to a stationary cart in the opposite direction to a 'falling' pendulum. All other initial parameters are given: the mass, the velocity and acceleration of the pendulum and the mass of the cart. We assume no friction in the wheel axles nor between the wheels and the track.
- Inverted Pendulum+Cart Behaviour #3:

  A force f'' is instantaneously applied to a stationary cart in the same direction as a 'falling' pendulum. All other initial parameters are given.
- ...

Pragmatic Models: *Das Ding für Mich*[13]

Each of the above inverted pendulum and cart behaviours, we claim, corresponds to a pragmatic model.

So the 'base model' serves as a reference model. The "goodness" of a (proposed) 'base model' is thus "measured" by the "ease" with which various 'pragmatic models' can be expressed as 'implementations' of that 'base model'. We leave the (*tongue-in-cheek*) double quoted terms undefined!

1.8 Discussion: The Infrastructure

1.8.1 The Question

So what is it, then, that technically, and not just socio-economically, makes the railway system sketched above an infrastructure ?

1.8.2 Towards an Answer

To answer that question we need to look a little more at our example.

There are a number of things that we have omitted to cover in our pragmatic models. They were implicit in the 'base' model.

These are:

- Distribution of Components & Processes:

  To Be Written

- Concurrency of Operations:

[13] And similarly: "the things as we see them" are here equated with the various "pragmatic models".

To Be Written

- Independence of Operations:

  To Be Written

- Multiple Levels of View (or Abstractions):

  To Be Written

- Locality of Modifications:

  To Be Written

1.8.3 A First Answer

Now the answer is made relatively more easy to state.

To Be Written

2 General Definitions

The term infrastructure is a relatively modern term[14]. It is more frequently used in socio--economic than in scientific, let alone computing science, contexts. Yet, if we are to develop software for "something" that our customers believe is an infrastructure (component), then we had better understand the term. Below we develop a kind of definition:

1. **World Bank: Infrastructure**

   According to The World Bank, infrastructure is *an 'umbrella' term for activities referred to as "social overhead capital"; ...* and *encompasses activities that share technical and economic features* (such as economies of scale and spill-overs from users to non-users).

2. **Technical: Infrastructures**

   The analysis of section 1.7 suggests, to us, that an infrastructure is *the composition of parallel and communicating processes whose behaviour allows orderly (timely, uncorrupted and economical) interaction between processes, incl. people. (i.e. synchronization and communication of messages, materials, etc.)*

3. **UNU/IIST's Approach:**

   Hence UUN/IIST's approach in the research and development of software support for infrastructures is *to enable designs that allow monitoring & controlling (ie. managing) any infrastructure like a business, and laying bare as many features of the infrastructure as technically possible so as to give users and other stake-holders a strong voice and real responsibility.*

[14] Winston Churchill is quoted as having said, in the House of Commons, in 1946: *...the young Labourite speaker, that we just heard, obviously wishes to impress his constituency with the fact that he has attended Eton and Oxford when he uses such modern terms as 'infrastructure' ...*

3 Examples of Infrastructures: Informal and Formal

3.1 ABC'2000: Airline Business Computing

3.1.1 Syntax & Semantics — Narrative & Formal Signatures

We first enumerate and characterize all the relevant airline business components:

- SY: The System [state] — consists of:
  1. RE: Resources: [i]
     (a) AC: Aircrafts [ii]
     (b) AS: Airline Staff [iii]
     (c) FI: Finances — Available (Liquid + Credit) Cash [iv]
     (d) FA: Facilities: [v]
        i. AV: Aircraft tractors [vi]
        ii. EQ: Cargo loading/unloading equipment [vii]
        iii. *&c.*
  2. MO: Manifest Objects: [viii]
     (a) F: Flight [ix]
     (b) TF: Air Traffic [x]
     (c) PS: Passengers [xi]
     (d) CA: Cargo [xii]
  3. BT: Base Tables: [xiii]
     (a) FT: Financial Tables [xiv]
     (b) ST: Statistics [xv]
     (c) *&c.*
  4. MP: Management Plans: [xvi]
     (a) TP: Target (Plan) [xvii]
     (b) RP: Resource Plan [xviii]
     (c) FP: Flight Plans [xix]
  5. OD: Operating Documents: [xx]

    (a) TT: Timetable(s) [xxi]
    (b) BK: Bookings [xxii]
    (c) GL: General ledger, Accounts, etc.

6. MD: Manifest Documents: [xxiii]
    (a) AT: Air Tickets [xxiv]
    (b) AB: Airway Bills [xxv]

type

$SY = RE \times MO \times BT \times MP \times OD \times MD$
$RE = AC \times AS \times FI \times FA$
$MO = F \times TF \times PS \times CA$
$BT = FT \times ST \times ...$
$MP = TP \times RP \times FP$
$OD = TT \times BK \times GL$
$MD = AT \times AB \times ...$
$FA = AV \times EQ \times ...$

Some general observations may be in order.

Documental Representations: The timetables (with their cost information) represent the basis for doing business. We consider a timetable — as we may hold it in our hand, or as it may be displayed on a board or on a computer screen — to be a syntactic object.

We assume that if a flight is listed in a timetable then the resources necessary for actually operating that flight are available and that appropriate resource and flight plans are available and being followed.

The bookings represent a commitment to do business and thus represents one kind of summary of [the state of] that business as it is dynamically unfolding.

The bookings imply actions. Depending on the time at which we look at a bookings list, some of these actions are assumed to have been done while others are yet to be done. Some actions may be in the process of being performed.

Passenger tickets and freight airway bills are dual forms of the bookings: for every ticket or bill an appropriate entry must be found in the bookings, and vice-versa: for every entry in the bookings list tickets, respectively bills must have been issued.

Physical Manifestations: Flights and hence air traffic represent another kind of [state] summary of the dynamics of the business.

Document vs. Physics Representations: Thus the physical evidence: flights and air traffic, and in fact all the minute actions being taking in for example readying an aircraft, loading it, dispatching it, flying it, etc., are all "results", so-to-speak, of timetable and booking entries.

The syntax of timetables and bookings, like the syntax of target plans, resource plans and flight plans, namely the documents, whether electronic or on paper, are *prescriptions for actions. By semantics we mean the description of these actions at various levels of abstraction.*

Resources: The aircraft, the airline staff, the actual finances (available cash and credit) etc., represent the resources.

Action Denotations: Target plans (TP), resource plans (RP) and flight plans (FP) represent three syntactic views of the airline business.

We may call these views the strategic (TP), the tactical (RP) and the operational (FP) views.

They cover the build up (or trimming down) of resources (TP), the allocation and scheduling of these resources (RP), and the dispatch, monitoring and control of the resources (FP).

Section 3.1.4 is devoted specifically to the pragmatics of the various components and processes on them.

The syntactic views (of the various plans) are matched by semantic views. We review two forms of semantic views, or just 'semantics'.

We can either explain the semantics mathematically or we can explain it operationally.

The mathematical (denotational) semantics (view) assigns a function (or relation) to a syntactic component.

The operational semantics (view) description is given in terms of the actions that are implied by "following the plans".

We now informally describe the components and the functions and operations over them. The narrative is accompanied by some sketches of the signatures of a formal model.

1. Aircraft: An aircraft is a resource. It has a type (A300, B747/400, etc.), has passenger, cargo and fuel capacity; has minimum, medium and maximum flying distance attributes, etc. It also has a status with respect to it being allocated and scheduled for flights, the state of its condition: fuel load, wear and tear, etc.

   type
   $AC = AC\_id \xrightarrow[m]{} (AC\_Typ \times AC\_Cap \times AC\_Status \times AC\_Fuel \times ...)$

   Functions on aircraft: {i} availability, {ii} location, {iii} fuel level, {iv} wear and tear, {v} current configuration, etc. Operations on aircraft: {vi} put into service, {vii} refuel, Behaviour over an aircraft: {viii} a flight — see below.

2. AS: Airline Staff: The airline staff resource is represented in terms of a record of all the personnel, their allocation [xxvi] and scheduling [xxvii] with respect to tasks etc.

   $AS = Staff\_id \xrightarrow[m]{} (Staff\_typ \times Staff\_CV \times Staff\_SchAlloc \times ...)$

   Functions and operations on staff include: {ix} surveying availability of staff, [re-]allocation [xxviii] and [re-]scheduling [xxix] of individual staff members to tasks, [xxx] recording of staff tasks initiated, under way, or completed, etc.

3. FI: Finances: We here refer to 'cash-on-hand', credits (and debits), etc.

type FI = cash:**Int** × cred:**Int** × deb:**Int** × ...

Functions and operations on finances include: {x} replenishing funds, {xi} withdrawing cash, {xii} committing funds, {xiii} establishing credit, etc.

4. FA: Facilities: We here refer to such ground resources as:

   (a) Accessible fuel
   (b) Accessible aircraft tractors
   (c) Luggage and cargo loading & unloading facilities
   (d) *&c.*

5. Flight: A flight [xxxi] is the enactment of a flight plan entry: from the time an aircraft (as designated by the flight plan) is put into service (for example leaves the maintenance area to become fueled), during the periods of being loaded at a gate, taxiing down the tarmac (apron), flying, etc., until it is again taken out of service.

   For each leg[15] of a flight, the flight contributes to the air traffic from the time the aircraft occupies the departure runway to the time it reaches the arrival runway.

   type
   $F = T \overset{\sim}{\rightarrow} F\_n \times F\_Info$
   $F\_Info = AC\_id \times Pos \times Ori \times AC\_Fuel \times Vel \times Acc \times ...$

   T refers to real time. Pos[ition] and Ori[entation] refer to spatial co-ordinates. Vel[ocity] and Acc[eleration] refer to dynamics. The ... refers to the state of other variables of the aircraft.

6. Air Traffic: Air traffic is the time-wise, spatial position and orientation [xxxiii] of flights in a certain geographical area.

   type $TF = T \overset{\sim}{\rightarrow} (F\_n \underset{m}{\rightarrow} F\_Info)$

   We were not listing a flight as a manifest object in the system state but rather the air traffic aggregation of all flights.

7. Passengers:

   To Be Written

8. Cargo:

   To Be Written

[15] A flight leg [xxxii] is the route between two airports served by an aircraft.

9. Financial Tables: Financial tables summarize cost figures common to the acquisition, operation, maintenance and disposal of material and human resources and to external and internal services. The financial tables, for example, list current salary tables for staff, including overtime payment rules; airport fees for various aircraft types; etc.

   Financial tables are constructed as you would expect them to be! Functions and operations upon them include {xiv} revision and {xv} invocation when constructing targets, resource plans and flight plans.

10. Statistics: Statistics record all aspects of past performance: resource utilization, bookings, financial movements, etc.

11. Target Plan: A target covers a certain period — from some start date to some end date; typically the period covers two or more years. The plan contains many entries. Each entry [xxxiv] specifies for each pair of airports (that the airline wishes to fly (or flies) between), [xxxv] estimated number of passengers, [xxxvi] estimated cargo weight (volume), [xxxvii] expected income, [xxxviii] operating costs and [xxxix] profit.

    At any one time an airline is working according to only one target (plan).

    Target plans are created {xvi} on the basis of past airline performance (including statistics), strategic government, board and management decisions, financial tables, etc. Other operations on targets are adding, modifying, deleting and inspecting entries. Functions on targets include {xvii} evaluating the status of a target with respect to the actual state of the airline business, and {xviii} creating a resource plan.

    A target plan denotes a function from resources and base tables to sets of resource and flight plans, namely the ones that all satisfy the targets of the target plan.

    $\textbf{type}\ [\mathrm{TP}]\text{: } (\mathrm{RE} \times \mathrm{BT}) \overset{\sim}{\rightarrow} (\mathrm{RP} \times \mathrm{FP})\textbf{-infset}$

12. Resource Plan: A resource plan specifies for each time interval a set of actions to be performed (on resources, etc.) by humans and otherwise so that the target can be reached. These actions are represented in terms of [xl] resource action descriptors. Other than illustrating some intended actions we do not further describe the variety of these descriptors.

    Mathematically a resource plan denotes a function from resources and timetables to sets of flight plans, namely the flight plans which, with the stated resources and timetables, satisfy the resource plan (and hence the target plan).

    $\textbf{type}\ [\mathrm{RP}]\text{: } (\mathrm{RE} \times \mathrm{TT}) \overset{\sim}{\rightarrow} \mathrm{FP}\textbf{-infset}$

    A resource plan operationally denotes a sequence of resource changes (etc.): some (new) resources need to be acquired, old ones taken out of service or sold, existing ones replenished (cash, fuel, spare parts) or substituted (staff), etc.[16]

[16] Later we shall see that a flight plan denotes a sequence of actions related primarily to flights, hence to a subset of the air traffic in the area covered by the flight plan.

Typical actions range from strategic acquisitions via tactical purchase to operational handling. Buying an aircraft is considered a strategic action. Purchasing fuel may be considered a tactical action. Fueling an aircraft is considered an operational action. We omit an exhaustive listing of all actions![17] The action of constructing a flight plan may be part of a resource plan.

Resource plans are constructed based on the current target plan and current resources. Other operations on resource plans include adding, modifying, deleting and inspecting entries. Functions on resource plans include {xix} evaluating the status of a resource plan with respect to the actual state of the airline business, and creating timetables and a flight plan.

13. Flight Plans: A flight plan consists of a set of flight descriptors. A flight descriptor gives a flight number, the flight route, aircraft readiness, preparation, and departure & arrival times (including gate arrival, passenger alighting & boarding, cargo unloading & loading, and gate closing times), summarizes information about the aircraft[18], expected financial parameters (cost, fees, passenger service and staff expenses, etc.; expected [and possibly actual] revenues, profit/loss), crew allocation, etc.

    A flight plan is constructed from the current timetable and the current resource plan.

    Operations on a flight plan include adding, modifying, deleting and inspecting an entry. Entries need to be inspected so as to run, monitor, and partially control the designated flights. Inspection is usually done by airline staff.

    Mathematically a flight plan is represented as a function from resources and manifest objects to resources and manifest objects.

    type [FP]: $(\mathrm{RE} \times \mathrm{MO}) \overset{\sim}{\to} (\mathrm{RE} \times \mathrm{MO})$

    A flight plan operationally denotes a sequence of actions that relate specifically to flights, including the traffic composed from those flights.

14. Timetable(s): A timetable covers a period (a [flight] season), and indicates for airports identities of air flights and arrival and departure times.

    type $\mathrm{TT} = \mathrm{F\_n} \ \overrightarrow{m}\ (\mathrm{A} \ \overrightarrow{m}\ (\mathrm{arr{:}}(\mathrm{Date} \times \mathrm{Time}) \times \mathrm{dep{:}}(\mathrm{Date} \times \mathrm{Time})))$

    Mathematically a timetable is represented as a set of air traffics.

    type [TT]: TF-**infset**

To Be Written

[17] We shall later consider some computerised form of describing and supporting the enactment of these actions. For the time being we are just dealing with the domain: not how it may be, or already is, computerised.

[18] — which is designated to perform, or which is performing, or which has already performed the flight referred to.

15. Reservations:

To Be Written

16. Air tickets:

To Be Written

17. Airway Bills:

To Be Written

3.1.2 A 'Base' Formal Model

We summarize the above fragments into a more coherent description:

3.1.3 Towards a Theory of Air Line Business

3.1.4 Pragmatics: Strategies, Tactics & Operations

Airline Specific Actions :

Market Planning :
Flight Planning :
Reservations & Sales :
Financial Operations :
Staff Operations :
Business Operations :

(Shared) Airport Operator Services :

Departure Control :
Passenger & Cargo Service :

Other (Shared) Airport *&c.* Services :

Maintenance Engineering :
Air Traffic Control :

3.1.5 The Pragmatics: A Formal Model

3.1.6 Discussion

3.2 ATC'2000: Air Traffic Control

3.2.1 Syntax & Semantics: A Narrative

1. Airspace

   Airports, Air-domes, Air Lanes/Corridors, Networks

   Opening/Closing

2. Timetables

   Airline, Airport, and Network Schedules

3. Traffic

   Scheduled vs. Real Traffic

   Scheduling and Re-scheduling

4. Air Traffic Control

   Monitoring & "Control": Dispatch, Arrival, En-route

3.2.2 A 'Base' Formal Model

type
  AS = inas:A-**set**
    $\times$ (A $\underset{m}{\rightarrow}$ (D $\times$ (A $\underset{m}{\rightarrow}$ W-**set**)))
    $\times$ ouas:A-**set**
  TT = F $\underset{m}{\rightarrow}$ (A $\underset{m}{\rightarrow}$ (T $\times$ T))
  sTF = T $\overset{\sim}{\rightarrow}$ (F $\underset{m}{\rightarrow}$ E)
  rTF = T $\overset{\sim}{\rightarrow}$ (F $\underset{m}{\rightarrow}$ E)
  D, W, E
value

  inv_AS: AS $\rightarrow$ **Bool**
  inv_TT: TT $\rightarrow$ **Bool**
  inv_TF: TF $\rightarrow$ **Bool**
  inv_TT_AS: TT$\times$AS $\rightarrow$ **Bool**
  inv_TF_TT: TF$\times$TT $\rightarrow$ **Bool**
  inv_TT_TT_AS: TT$\times$TF$\times$AS $\rightarrow$ **Bool**, ...

  disruption: rTF $\times$ sTF $\times$ ((rTF $\times$ sTF) $\rightarrow$ **Bool**)-**set** $\rightarrow$ T

  Open_A: A$\times$AS $\overset{\sim}{\rightarrow}$ AS, Close_A: A$\times$AS $\overset{\sim}{\rightarrow}$ AS
  Open_W: W$\times$(A$\times$A)>AS $\overset{\sim}{\rightarrow}$ AS
  Close_W: W$\times$(A$\times$A)>AS $\overset{\sim}{\rightarrow}$ AS
  Schedule: AS$\times$TT $\overset{\sim}{\rightarrow}$ sTF-**set**
  BestSchedule: AS$\times$TT $\overset{\sim}{\rightarrow}$ sTF, ...

3.2.3 Towards a Theory of Air Traffic

3.2.4 Pragmatics: Air Traffic Control

3.2.5 An ‘Air Traffic Control’ Model

3.2.6 Discussion

3.3 MoFIT: Ministry of Finance IT

3.3.1 Syntax & Semantics: A Narrative

1. Accounts

   Credits, Debits, Balance, Statements, . . .

2. Transactions

   Receipts, Payments, Transfers: In/Out, . . .

3. Budgets

   Simple and Compound: Hierarchical & Distributed

4. Policies: Rules & Regulations

   Enforceable, Reportable

   Transaction Authorization, resp. Exceeding Credit Limits, . . .

5. Planning & Budgeting

   Estimation of Income, resp. Expenditure

   Distribution over Budgets

   [Re-]Formulation/Review of Policies & Plans

6. Audit & Security

7. Income [Tax, Duty, *&c.*] Collection

8. Funds Disbursement/Transfer

3.3.2 A ‘Base’ Formal Model

```
type
   Acct, Staff, Data, Time
   Cmd =  Paym | Rcpt |Bal | In | Out | Stmt | ...
   Sum  = Int
value
   credit: Acct × Sum × Staff → Acct
   debit:  Acct × Sum × Staff → Acct
   balnc:  Acct × Staff → Sum
   xfer:   Acct × Acct × Date × Sum × Staff
                 → (Acct × Acct)
   stmt:   Acct × Staff → (Cmd × Sum)*
```

```
type
   Bn
   Cmd′ = Cmd | Accts | ...
   Budget = Bn $\underset{m}{\rightarrow}$ (Acct | Budget)
value
   accts: Budget × Staff → Acct-set
   MoF: Budget
```

```
type
   Rn
   Cmd″ = Cmd′ | Rept
   Rept = Rn
   Policies = Enforceable | Reportable
   Enforceable = (MoF → Bool)-set
   Reportable  = Rn $\underset{m}{\rightarrow}$ (MoF $\xrightarrow{\sim}$ Report)
   Report = (Staff × Date × Time × Cmd″)$^*$
value

   inv_MoF: MoF × Enforceable → Bool
   inv_MoF(mof,enf) ≡
      ∀ p:Enforceable • p ∈ enf ⇒ p(mof)

   grept: MoF × Reportable $\xrightarrow{\sim}$ Report-set
   grept(mof,rpt) ≡ { r(mof) | r:Reportable • r ∈ rng rpt }

   srept: MoF × Reportable × Rn $\xrightarrow{\sim}$ Report
   srept(mof,rpt,rn) ≡ (rpt(rn))(mof)
   pre: rn ∈ dom rept
```

3.3.3 Towards a Theory of Budgets and Accounts

3.3.4 Pragmatics: Distributed Collection and Disbursal

3.3.5 A ‘Distribution’ Model

3.3.6 Discussion

3.4 MI^{2}CI: Manufacturing Systems

3.4.1 Syntax and Semantics: A Narrative

3.4.2 A ‘Base’ Formal Model

```
type
   WFS = (C $\underset{m}{\rightarrow}$ g:(A $\underset{m}{\rightarrow}$ (nxt:(A $\underset{m}{\rightarrow}$ E-set)
                                          × xin:E-set
                                          × xout:E-set
                                          × cell:W
```

```
                                  $\times$ process:((V-**set** $\overset{\sim}{\rightarrow}$ V-**set**)
                                  $\times$ Priority
                                  $\times$ Cost))
              $\times$ (W $\overrightarrow{m}$ (stg:(E $\overrightarrow{m}$ V-**set**) $\times$ tray:(E $\overrightarrow{m}$ V-**set**)))))
    Priority, Cost, Price, Profit = **Nat**
**value**

    queues: WFS $\rightarrow$ (W $\overrightarrow{m}$ (A $\overrightarrow{m}$ **Nat**))
    idle: WFS $\rightarrow$ (W $\overrightarrow{m}$ A-**set**)
    cost: WFS $\rightarrow$ (C $\overrightarrow{m}$ Cost)
    profit: WFS $\times$ C $\times$ Price $\overset{\sim}{\rightarrow}$ Profit

    insert_C: WFS $\times$ C $\times$ G $\overset{\sim}{\rightarrow}$ WFS
    delete_C: WFS $\times$ C $\overset{\sim}{\rightarrow}$ WFS
    [re$-$]sched: WFS $\times$ (C $\overrightarrow{m}$ (A $\overrightarrow{m}$ Priority)) $\overset{\sim}{\rightarrow}$ WFS

    next: WFS $\times$ (A $\overrightarrow{m}$ V-**set**) $\overset{\sim}{\rightarrow}$ (WFS $\times$ (A $\overrightarrow{m}$ V-**set**))
```

3.4.3 Towards a Theory of Manufacturing Industries

3.4.4 Pragmatics: Business & Management Practices

Strategic, Tactical & Operational "Supports"

1. Inter-Market

   Relations between Industry, Services, Banking, etc. Sectors

   What & How: goods, services, etc. are offered;

   Where [Trends]: Economic, products, services, etc.

   Why: Policies, Rules & Regulations, Laws

2. Intra-Industry Sector/Inter-Enterprises

   Relations between Enterprises: Buying/Selling

3. Intra-Enterprise/Inter-Division

   Relations between Enterprise Divisions:

   New Products Development "Cycle"

   Order/Manufacturing "Cycle", CIM, FMS, ...

4. Intra-Division

   Relations between Cells: CAD, CAE, CAM, ...

3.4.5 Pragmatics: Business & Management Practices

3.4.6 Discussion

3.5 Multi-Script

3.5.1 Syntax & Semantics: A Narrative

1. Core Model:
   (a) Documents as Frames
   (b) Frame "text" having Orientation, Metrics, and a list of Strings
   (c) Strings being lists of Tokens
   (d) Tokens being either Characters or Frames
2. Document Models:
   (a) *Book*
      i. Books being lists of Chapters
      ii. Chapter having a Display Line, Text and a list of
   (b) *Newspaper:*
      i. A Newspaper as a list of Pages
      ii. Pages with Heads, Foots and Column Blocks
      iii. Single/Multiple Column Blocks
      iv. S/M Column Texts, Photos, ...
      v. Inserts
      vi. *&c.*
   (c) *Administrative Form:*
      i. Head
      ii. Fields: Title, Fill-in Text, Help Text
      iii. Foot
   (d) *Letter*
   (e) *&c.*
3. Format Information:
   (a) Page size
   (b) Text Width, Height
   (c) Margin Parameters
   (d) *&c.*
4. Transformation Functions:
   (a) Core document ↔ Book
   (b) Core document ↔ Newspaper
   (c) Core document ↔ Form
   (d) Core document ↔ Letter

3.5.2 A 'Base' Formal Model

```
type
   $D = F$
   $F = O \times M \times S^*$
   $S = T^*$
   $T = C \mid F$

value
   ID: $D \overset{\sim}{\to} (\mathbf{Nat} \times \mathbf{Nat})$
   IF: $F \overset{\sim}{\to} (\mathbf{Nat} \times \mathbf{Nat}) \overset{\sim}{\to} (\mathbf{Nat} \times \mathbf{Nat})$
   IS: $O \times M \times S \overset{\sim}{\to} (\mathbf{Nat} \times \mathbf{Nat}) \overset{\sim}{\to} (\mathbf{Nat} \times \mathbf{Nat})$
   ITl: $O \times M \times T^* \overset{\sim}{\to} (\mathbf{Nat} \times \mathbf{Nat}) \overset{\sim}{\to} (\mathbf{Nat} \times \mathbf{Nat})$
   IT: $O \times M \times T \overset{\sim}{\to} (\mathbf{Nat} \times \mathbf{Nat}) \overset{\sim}{\to} (\mathbf{Nat} \times \mathbf{Nat})$

   ID(d) $\equiv$ IF(d)(0,0)

   IF(o,m,sl)(x,y) $\equiv$
      case sl:
         <> $\to$ MF(o)(x,y)
         _ $\to$ ISl(o,m,tl sl)IS(o,m,hd sl)(x,y)
      end
```

```
type
    Book = Chap$^*$
    Chap = Displ $\times$ Txt (Sect $\times$ Text)$^*$
    Sect = Displ $\times$ Txt (Subs $\times$ Text)$^*$
    SubS = ...

    BookForm
    NewsForm

value
    inv_Book: D $\to$ Bool
    Doc_Book: D $\overset{\sim}{\to}$ Book
    Book_Doc: Book $\to$ D
    FormatBk: Book $\times$ BookForm $\overset{\sim}{\to}$ Print

    inv_News: D $\to$ Bool
    Doc_News: D $\overset{\sim}{\to}$ News
    News_Doc: News $\to$ D
    FormatNw: News $\times$ NewsForm $\overset{\sim}{\to}$ Print
```

3.5.3 Towards a Theory of Multi-lingual Documents

3.5.4 Pragmatics: Text Interchange

3.5.5 Pragmatics: A Formal Model

3.5.6 Discussion

3.6 Worldwide GIS *&c.* Information Infrastructures

3.6.1 Syntax and Semantics: A Narrative

3.6.2 A ‘Base’ Formal Model

```
type
   D, Txt, T, V, A, N, R

   GaDIIS = S × Ψ

   S = (D $\underset{m}{\rightarrow}$ (domtxt:Txt ×
             ((T $\underset{m}{\rightarrow}$ (typtxt:Txt ×
                (((V $\underset{m}{\rightarrow}$ (vertxt:Txt × (A × W)))
                            × S)) × UO) × NO)) × Env × Σ))

   Ψ = A $\underset{m}{\rightarrow}$ I
   W = I $\overset{\sim}{\rightarrow}$ I

   ρ:Env = N $\underset{m}{\rightarrow}$ ((D×T×V)|(D×(T×V)$^*$)|(D×T×V)$^*$)
   σ:Σ = N $\underset{m}{\rightarrow}$ (T$^*$ × T$^*$ × T) × R

   or:

   `Rho = N $\underset{m}{\rightarrow}$ L × ((D×T×V)|(D×(T×V)$^*$)|(D×T×V)$^*$)
   Σ = L $\underset{m}{\rightarrow}$ R

   UO = Un $\underset{m}{\rightarrow}$ (fcttxt:Txt ×
          (parm:T$^*$ × i:T × r:T) × (((I$^*$ × I)|Txt) $\overset{\sim}{\rightarrow}$ R))

   NO = Nn $\underset{m}{\rightarrow}$ (fcttxt:Txt ×
          (parm:T$^*$ × il:T$^*$ × r:T) × (((I$^*$ × I$^*$)|Txt) $\overset{\sim}{\rightarrow}$ R))
```

3.6.3 Towards a Theory of GIS + DIS Infrastructures

3.6.4 Pragmatics: Decision Support Systems for Planning

3.6.5 Pragmatics: A Formal Model

3.6.6 Discussion

3.7 Library Information, Monitoring and Command System

3.7.1 Syntax and Semantics: A Narrative

1. The System:

   The library system consists of a number of libraries, a number of document providers (publishers, book distributors, book stores, etc.), and a number of borrowers. In any reality they can get to know about each other through a means that we shall here call a 'trader'[19].

   type
   L, P, B
   $\Sigma = (L \xrightarrow[m]{} LIB) \times (P \xrightarrow[m]{} PBL) \times (B \xrightarrow[m]{} BRW) \times TRDR$

   Operations on the overall system state include those of establishing, operating and disbanding libraries, document providers, borrowers, etc.

   value
   est_Lib: $(L \times LIB) \stackrel{\sim}{\rightarrow} \Sigma \stackrel{\sim}{\rightarrow} \Sigma$
   op_Lib: $(L \times Lib\_Cmd) \stackrel{\sim}{\rightarrow} \Sigma \stackrel{\sim}{\rightarrow} \Sigma$
   dis_Lib: $L \stackrel{\sim}{\rightarrow} \Sigma \stackrel{\sim}{\rightarrow} \Sigma$
   est_Pbl: $(P \times PBL) \stackrel{\sim}{\rightarrow} \Sigma \stackrel{\sim}{\rightarrow} \Sigma$
   op_Pbl: $(L \times Pbl\_Cmd) \stackrel{\sim}{\rightarrow} \Sigma \stackrel{\sim}{\rightarrow} \Sigma$
   dis_Pbl: $P \stackrel{\sim}{\rightarrow} \Sigma \stackrel{\sim}{\rightarrow} \Sigma$
   est_Brw: $(B \times BRW) \stackrel{\sim}{\rightarrow} \Sigma \stackrel{\sim}{\rightarrow} \Sigma$
   op_Brw: $(B \times Brw\_Cmd) \stackrel{\sim}{\rightarrow} \Sigma \stackrel{\sim}{\rightarrow} \Sigma$
   dis_Brw: $B \stackrel{\sim}{\rightarrow} \Sigma \stackrel{\sim}{\rightarrow} \Sigma$

   The essence of operation (modelled here as the "execution" of appropriate commands) is described later. They are here thought of as "system-wide" operations as any one "command execution" may affect several parts of the system via the trader.

2. Libraries:

   A library consists of the following components[20]: ([i]) the documents[21], ([ii]) borrowers library cards, ([iii]) the catalogue(s) [listing all documents], ([iv]) on order (acquisition)

[19] In its classical form the 'trader' is implemented by a "network" of means of getting to know about the existence of one another: advertisements in journals, entries in telephone directories, membership of associations, etc.

[20] Components or sub-components, if represented in software, typically become state variables and are often retained as [part of] a database.

[21] Books, reports, journals, etc.

list [for new documents], ([v]) subscription (circulation) lists [typically for journals], ([vi]) document availability list [available, checked out, reserved, etc.], ([vii]) publication lists, ([viii]) borrower profiles, ... (etc.), and ([ix]) the accounts (including budget).

type

$$\begin{array}{l}
\quad \mathrm{D, Date, DDescr, DDescr', DDescr'',} \\
\quad \mathrm{B, BInfo, BDescr} \\
\\
\quad \mathrm{LIB = LDOCS \times CRDS \times CTLG \times ACQL \times SLST \times AVLL \times PUBL \times PROF \times ACCT} \\
\\
\quad \mathrm{LDOCS = D} \overrightarrow{m} \mathbf{Nat} \\
\quad \mathrm{CRDS = B} \overrightarrow{m} \mathrm{BInfo} \\
\quad \mathrm{CTLG = D} \overrightarrow{m} \mathrm{DDescr} \\
\quad \mathrm{ACQL = D} \overrightarrow{m} \mathrm{DDescr'} \\
\quad \mathrm{SLST = D} \overrightarrow{m} \mathrm{B^*} \\
\quad \mathrm{AVLL = D} \overrightarrow{m} \mathrm{B^*} \\
\quad \mathrm{PUBL = D} \overrightarrow{m} \mathrm{DDescr''} \\
\quad \mathrm{PROF = BDescr} \\
\quad \mathrm{ACCT = BUDG \times COMM \times PAID} \\
\quad \mathrm{COMM, PAID =} \mathbf{Nat}
\end{array}$$

For each component, and for a number of combinations of components the library allows a number of functions[22] and operations[23].

Example functions are: ({i}) search a catalogue for a document of a given topic; ({ii}) check whether a document is due for return; etc.

Example operations are: ({iii}) order a document [enter into acquisition list]; ({iv}) check out a document to a borrower [modify document availability list, etc.]; ({v}) receive of a document from a provider [move modified entry from acquisition list to catalogue]; ({vi}) "shred" a document; etc.

Some functions (like catalogue search) and operations may involve two or more libraries. Others (like ordering a document) may involve a library and a document provider. Checking out a book is an operation that involves a library and a borrower.

3. Document Providers:

   The term document provider refers to the ensemble of publishers, book (etc.) distributors, book stores, etc.

   We focus only on those aspects of a document provider which relate to libraries. Components are: ([x]) document source lists [similar to item ([xi]) above; e.g. publication lists]; ([xii]) library mailing lists and accounts; ([xiii]) inventory list; ([xiv]) and actual documents.

[22] Functions inspect the local state but do not change it.

[23] Operations may change the local state, i.e. the value of state variables. Some functions may change the global system state!

type

$$\begin{array}{l}
\text{DDescr}''', \text{PDOCS} \\
\text{PBL} \;\;= \text{PBLL} \times \text{LIBL} \times \text{INVL} \times \text{PDOCS} \\
\text{PBLL} = \text{D} \overrightarrow{m} \text{DDescr}''' \\
\text{LIBL} = \text{L} \overrightarrow{m} \text{LibDescr} \times (\text{COMM} \times \text{PAID}) \\
\text{INVL} = \text{D} \overrightarrow{m} \mathbf{Nat}
\end{array}$$

Example functions are: ({vii}) responding to an inquiry as to availability and price of a set of documents; ({viii}) sending publication lists to libraries, etc.

Example operations are: ({ix}) accepting an order; ({x}) delivering an order; ({xi}) sending an invoice; ({xii}) receiving payment, ({xiii}–{xiv}) [re-]printing or “shredding” documents and entering them into or removing them from storage and the inventory [list]; etc.

4. Borrowers:

   Components are: ([xv]) documents borrowed and ([xvi]) documents on order.

   Example functions and operations: ({xv}) inquire about a document, . . . , etc.; ({xvi}) check out a book; ({xvii}) return a book; etc.

   type

$$\begin{array}{l}
\text{Date} \\
\text{BRW} = \text{BDOCS} \times \text{OORD} \\
\text{BDOCS} = \text{D} \overrightarrow{m} (\text{Date} \times \text{DOC}) \\
\text{OORD} = \text{D} \overrightarrow{m} \text{Date}
\end{array}$$

5. Trader:

   The trader components could be ([xvii]) lists of libraries (with their specialities), ([xviii]) lists of document providers (etc.), and even ([xix]) lists of borrowers (with their interest profiles).

   type

$$\begin{array}{l}
\text{LibDescr, PublDescr, BorrDescr} \\
\text{TRDR} = \; (\text{L} \overrightarrow{m} \text{LibDescr}) \times (\text{P} \overrightarrow{m} \text{PublDescr}) \times (\text{B} \overrightarrow{m} \text{BorrDescr})
\end{array}$$

   Example functions are: respond to inquiries from libraries ({xviii}), document providers ({xix}) or borrowers ({xx}). Some inquiry functions may change the state of the inquirer.

   Example operations are: ({xxi}–{xxii}–{xxiii}) enter, ({xxiv}–{xxv}–{xxvi}) modify and ({xxvii}–{xxviii}–{xxix}) delete library, publisher and borrower entries.

3.7.2 A ‘Base’ Formal Model

We “complete” the model by defining some of the functions whose signatures we have already given.

3.7.3 Towards a Theory of Library Systems

Laws express properties that hold irrespective of the effect of the execution of otherwise validly expressed functions and operations across an entire system of libraries, document providers and borrowers.

Before detailing some of these laws we briefly give an example:

1. Law(s) of Conservation:

   The total number of documents in the system is unchanged between two consecutive document printings and/or document "shreddings".

Any implementation of any software requirements should result in a system (the above + the software) that obeys these laws.

Other examples of laws are:

2. :
3. :
4. :
5. :
6. :

3.7.4 Pragmatics: Distribution and Federated Trading

3.7.5 Pragmatics: A Formal Model of Distribution and Federated Trading

3.7.6 Discussion

4 A Methodological Setting

4.1 Domain Analysis

4.2 Requirements Capture

4.3 [Systems &] Software Architecture

4.4 Software Design

5 Some Research Challenges

6 Some Societal Challenges

7 Conclusion

8 Bibliography

References

[1] Derek Andrews. *The VDM-SL Standard.* ISO (International Standards Organisation, 1994. This is the offical VDM Specification Language Standard.

[2] Dines Bjørner, Chris W. George, and Søren Prehn. Domain Analysis — a Prerequisite for Requirements Capture. Technical Report UNU/IIST Technical Report No. 37, UNU/IIST, P.O.Box 3058, Macau, March 1995.

[3] Dines Bjørner and Cliff B. Jones. *VDM: The Meta-Language.* Lecture Notes in Computer Science. Springer-Verlag, 1978.

[4] Dines Bjørner and Cliff B. Jones. *Formal Specification and Software Development.* C.A.R. Hoare Series in Computer Science. Prentice Hall Intl., 1982.

[5] Chris W. George et al. *The RAISE Specification Language.* BCS Practititioners Series. Prentice Hall, 1992.

[6] Chris W. George et al. *The RAISE Method.* BCS Practititioners Series. Prentice Hall, 1995.