PERFORMANCE ANALYSIS FOR SYSTEM SELECTION

Chandru Mirchandani
Lockheed-Martin Space Operations,

Parminder Ghuman
NASA/GSFC Microelectronics & Signal Processing Division,
NASA Goddard Space Flight Center, Greenbelt, Maryland 20771

ABSTRACT

The development of unique solutions to telemetry processing using the latest technologies is often fraught with the uncertainty of the system working correctly within the schedule for operational support. This uncertainty can be reduced considerably by analyzing the performance of the system during the development and incremental test stage. This paper describes a method by which the analysis may be carried out during development so that the system will have the capability in the required time for mission support. This paper will show how different system models lend themselves to the requirements, and how the analyses identifies areas of high risk. The paper will also describe a case study whereby these three alternatives to telemetry processing have been used and could have been analyzed so that they would have met the requirements in a timely manner.

METHODOLOGY

Systems are aimed at meeting an objective within specified criteria. In the commercial world this criteria is schedule driven, whereas in research and development, it is cost driven. Whether a new technology is the valid choice for implementing a function depends on various factors. The methodology outlined in this paper identifies the following:

Functionality: The system should meet all the task-oriented requirements that it was designed for. Depending on the time, cost and complexity of the system these may not have been met completely. To measure the completeness of the design, we have to assume that (a) the system is critical to the project (b) the loss of data is not acceptable. If these assumptions are made, then we can quantify the requirements and the percentage of completeness on a weight scale.

Flexibility: The flexibility measure of performance depends on the prediction for the future needs of this type of system and its enhanced needs. To build in flexibility, the following factors have to be considered. How testable is this capability? How easy is it to re-engineer the design to meet the new needs? And can this system implementation be expandable to meet new requirements, predicted or otherwise. The quantifiable measures being the ease of testability, completeness of the test processes and how repeatable is the test process. The other two criteria, re-engineering and the ability to meet new requirements are quantifiable with cost, and schedule, which is directly proportional to the ease of implementation. Similarly, these measures depend on and vary with the type of technology; the complexity of the design; the expertise to achieve the implementation; the state-of-the-art of the technology to be used, and cost, schedule and information.

Reliability & Maintainability: All the traditional aspects of reliability, modeling and prediction have to be considered. The most difficult problem to be faced being: the knowledge base. The functional reliability depends on whether the objectives are met and how well and whether they are repeatable. The performance aspect is the quality and timeliness of the functions being performed and the lifetime of the system. The criteria and the wide disparity in its measures depend on the technology used. The complexity of the design; the expertise (experience) of the designer and his/her skill/information base; the schedule for the element, i.e. when is it needed and are adequate skills and funding available to implement a 'reliable good' system; and finally the information available. The last dependency is by far the most influential in arriving at realistic predictions. Newer and lesser mature technology has a very small database of performance. The number of sources (vendors) and the validity of their claims becomes very subjective and requires, in an ideal case, intense prototyping efforts; but as a raw measure, some good experiences and educated predictions.

Cost: As with all projects, cost is a very important measure in deciding how one is to implement the design. It should meet cost and scheduling objectives and do so without expending too much (funds) in acquiring the skills. As shown, the dependencies follow the classic pattern of 'how' and 'when' the system is implemented.

ANALYSIS

The objective is to analyze three systems A, B and C which have the same functionality but vary in performance with respect to rate of data processed, volume of data processed and quality of data processed. The analyses can be carried out for all the sub-systems for three systems A, B and C. To ensure that the comparison is consistent, it is imperative to state the objective of each subsystem and element within the subsystem. To simplify the comparison, the subsystems of A, B and C are classified into one of these three models.

Mixed System Model: The Mixed System model has modular elements as cards in a CPU cage with Back-Plane (B/P) interconnects, and a Local Area Network (LAN).

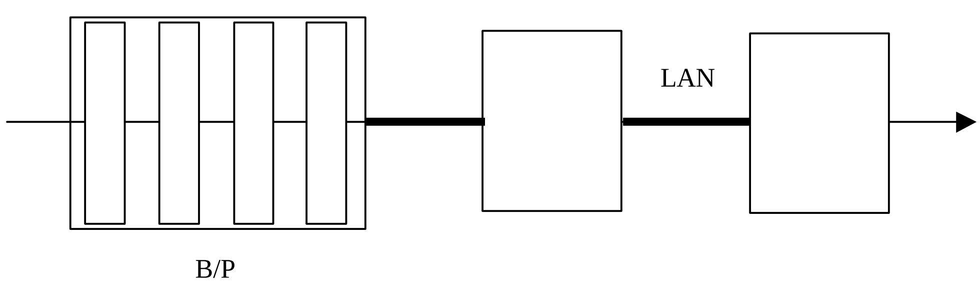

Figure 1 Mixed System Model

System A fits into this model because it uses legacy and new firmware and software elements, together with some hardware within the subsystems. The Front End System (FES-A) is depicted as block ‘A1’ in Figure 2.

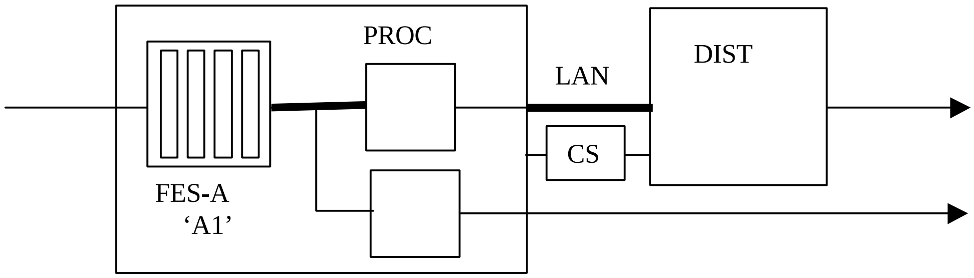

Figure 2 System A Block Model

Distributed System: The Distributed System model, with CPU intensive distributed processing functions. The interfaces between the elements are a Wide Area Network (WAN), local wire interconnects (I/F), and a LAN.

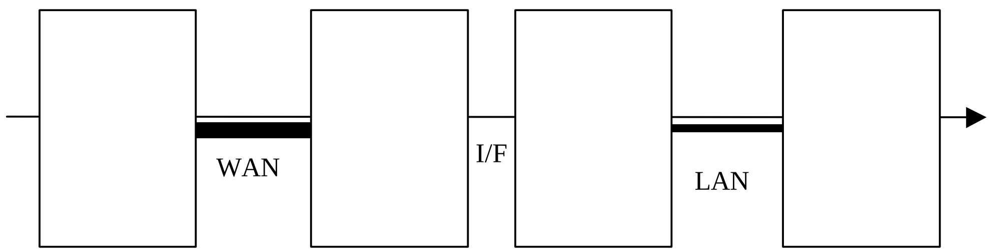

Figure 3 Distributed System Model

System B fits into this category because it utilizes CPU-intensive distributed processing functions. Once again the LZP Data Capture Front-End System, DCF-B is denoted as ‘B1’ in the Figure 4.

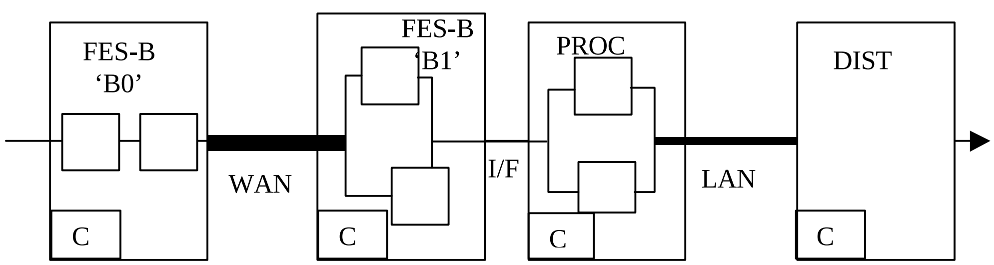

Figure 4 System B Block Model

Single Element System: The single card system with elements on chip level, in this case the interconnects are board interconnects which are Metal Traces (M/T), and a local wire interconnect (I/F) to the next element, Figure 5. System C fits into this category because it has all the front-end and processing functions in a single card and the distribution functions in the same host system. Figure 6, Block C1 denotes the Front-End System FES-C.

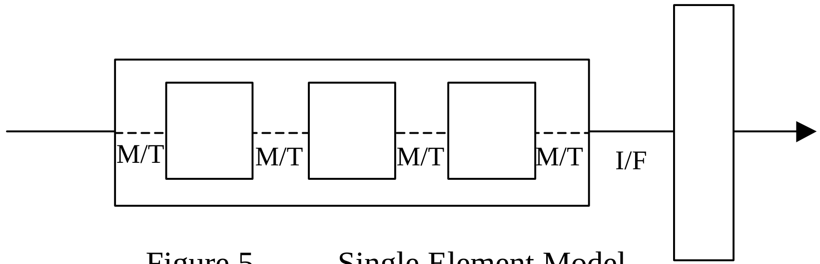

Figure 5 Single Element Model

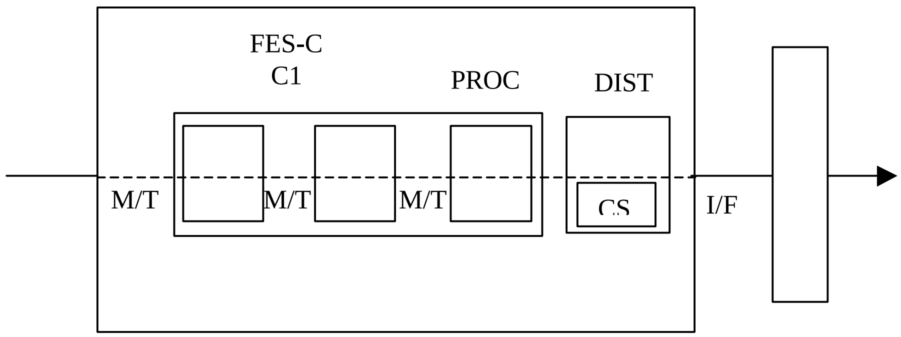

Figure 6 System C Block Model

Using these system level classifications, the analyses of the three systems can now be completed. It was evident from task listings that System B had many more points of failure than A or C, purely based on the number of separate elements. Using the same criteria and line of questioning, the rankings for the Front-End, Processing and Distribution systems are evaluated on a sub-system basis. The basis for ranking is on the type of system and represents a relative value. The Performance Measures are tailored towards the Performance Factors metrics.

In evaluating the *Individual Rankings* and *Average Rankings* for the sub-system elements for the Processing and Distribution Systems, the author has taken the liberty of using a relative comparison scheme due to a lack of expert opinion. Wherever possible the geometric mean was used to reduce the subjective effect of *Ranking*. Table 1 shows the *Scores* from the *Average Rankings*.

Using ranking calculations based on criticality, each of the major sub-systems FRONT-END, PROCESSING and DISTRIBUTION SYSTEMS for A, B, & C are evaluated, and subsequently the Overall Evaluations are calculated for the systems. Given these evaluations the Evaluation Criteria are pair-wise compared to determine the relative priorities of the criteria. Higher the priority, greater the need to ensure that the criteria is met to the fullest. This priority scheme in conjunction with the *ranking* determines the order in which the elements have to be optimally developed to ensure the optimal solution.

For example, the relative priority (RP) for System A FRONT-END Reliability & Maintainability is the highest, say 0.5, and the *Average Ranking* for the Functional and Performance Reliability & Maintainability of System A FRONT-END is, say 4.09. This would give a *Risk Score* (4.09 x 0.5) of 2.045. To lower this value, the element level scores are examined and the high value scores are identified.

Table 1 A, B, & C Front-End, Processing & Distribution Systems Rankings

SYSTEM A		SYSTEM B		SYSTEM C	
FRONT-END	SCORE	FRONT-END	SCORE	FRONT-END	SCORE
Functionality	1.62	Functionality	1.85	Functionality	1.54
Flexibility	3.85	Flexibility	3.93	Flexibility	1.55
Rel. & Maint.	4.09	Rel. & Maint.	4.17	Rel. & Maint.	1.48
Cost/Schedule	2.35	Cost/Schedule	1.95	Cost/Schedule	1.67
PROCESSING	SCORE	PROCESSING	SCORE	PROCESSING	SCORE
Functionality	2.28	Functionality	1.62	Functionality	1.73
Flexibility	3.51	Flexibility	4.18	Flexibility	1.68
Rel. & Maint.	6.05	Rel. & Maint.	4.84	Rel. & Maint.	1.42
Cost/Schedule	3.19	Cost/Schedule	1.80	Cost/Schedule	1.59
DISTRIBUTION	SCORE	DISTRIBUTION	SCORE	DISTRIBUTION	SCORE
Functionality	1.51	Functionality	2.43	Functionality	1.84
Flexibility	2.20	Flexibility	5.79	Flexibility	2.31
Rel. & Maint.	2.10	Rel. & Maint.	4.97	Rel. & Maint.	2.06
Cost/Schedule	1.54	Cost/Schedule	2.93	Cost/Schedule	2.50

Table 2 Reliability & Maintainability Ranking for FES-A Element (d)

Functional Reliability & Maintainability	Tech 0.166	Comp 0.278	Exper 0.278	Sched 0.111	Info 0.166	Score
(d) Transfer data from Ingest to detect/ decode/correct	9	4	8	7	8	
	1.494	1.112	2.224	0.777	1.328	6.94
Performance Reliability & Maintainability	Tech 0.313	Comp 0.313	Exper 0.188	Sched 0.063	Info 0.125	Score
(d) Transfer data from Ingest to detect/ decode/correct	9	9	8	2	5	
	2.817	2.817	1.504	0.126	0.625	7.89

In this case the element (d) showed values of 6.94 and 7.89 for the Functional and Performance Reliability & Maintainability *Ranking*. The *Average Ranking* for **Element (d)** in the System A FRONT-END Reliability & Maintainability is $\sqrt{}(6.94 \times 7.89)$, i.e. 7.4. This would give a *Risk Score* (7.4 x 0.5) of 3.7. The system models lend themselves to different options for reducing this value.

For example, in System A, the high *Risk Score* is attributed to the high values for the *Ranking* by *Factors.* In the case of **Element (d)**, the *Rankings* are shown in Table 2. To improve the *Risk Score*, it is necessary to reduce the *Ranking* in the high valued *Factors*.

For Functional Reliability & Maintainability Improvement:

- Improve the Technology Factor, through improved *Design Approach;* improved validation of the *New Modules*; or limit the *Number of Modules*. The latter may be constrained due to design selection. One method of validating the *New Modules* is through *Incremental Prototyping.*

- Improve the Expertise Factor, through training and skill enhancement. The constraint of having an in-experienced design team should be a critical influencing criterion for design selection. In the absence of choice, the only validation process for the design process is increased verification at the elemental level, i.e. *Incrementally Prototyping.*

- Improve the quality of information. This is a learning process, and in the absence of information, the design has to be prototyped at every design plateau to build on the capabilities, i.e. *Incrementally Prototyped.*

For Performance Reliability & Maintainability Improvement:

- Improve the Technology Factor, through improved *Design Approach;* improved validation of the *New Modules*; or *Prototyping*. One method of validating the performance of *New Modules* is through *Incremental Prototyping.*

- Improve the Complexity Factor, through ensuring that the critical elements are reliable within the performance requirements. The best way to ensure this is through *Incremental Prototyping* at the element and sometimes the sub-element level. Once the critical elements are functioning reliably, the validation is extended to the higher levels of the hierarchy.

- Finally, the Expertise Factor, as for the Functional Reliability & Maintainability, is based on the available expertise.

PERFORMANCE RESULTS FROM CASE STUDY

The following paragraphs describe the actual results from system test, integration test, and acceptance test of three systems that fit the models. In some cases the results are from actual operational scenarios.

*System A – Performance Measurement and Model Evaluation***:** Table 3 shows the measurements of System A Real Time transfer performance. This test did not attempt to stress the system beyond the requirements for the first three missions, resulting in lower measured rates for the last release.

To demonstrate the application of this model, the Front End System for System A was evaluated as shown in Table 4.

Table 3 System A - Real Time Data Transfer Performance -- Measured

System Requirements			Mission Support	Measured Values		
Requirement	System	RTOS		Release 0	Release 1	Release 2
Peak-single	10 Mbps	512 Kbps now 2 Mbps future	1 Mbps	1.2 Mbps	1.0 Mbps	1.3 Mbps
Peak-24 streams	20 Mbps	1.5 Mbps now 10 Mbps future	2.14 Mbps	1.2 Mbps	2 Mbps-3 streams	1.2 Mbps-3 streams
Peak-1 R/T	512 Kbps	512 Kbps now 1 Mbps future	850 Kbps	875 Kbps	830 Kbps	456 Kbps
Peak-24 R/T	512 Kbps	512 Kbps now 1 Mbps future	1.8 Mbps	N/A	2.1 Mbps-5 users	500 Kbps
Allowable R/T delay	3 sec	1 sec	-	-	-	Exceeded 16 of 18 sessions

Table 4 Front End System Model Evaluation

Function	Element	System	Rank
a) Ingest data correctly	100	100	1.0
b) Synchronize data stream	90	80	3.0
c) Maintain quality statistics	30	20	9.0
d) Transfer data from Ingest to detect/ decode/correct	100	90	2.0
e) CRC Decoding/Detecting	100	100	1.0
f) Reed Solomon Decoding/Detecting	100	100	1.0
g) Reed Solomon Correction	100	100	1.0
h) Deliver data from detect/decode/correct	20	10	10.0
			3.50

System B – Performance Measurement and Model Evaluation: Some of the information contained in this section has been obtained from the web site, from discussions with System Configuration Manager, System Software Engineer, Project Management Personnel and System Hardware Engineers. Some of the more critical problems are as follows:

1. Bit Slip in the front end ingest which may be caused by incorrect timing design.
2. Reduced capability in ingesting and processing data in Real Time.
3. Reduced capability in providing real time telemetry data packets.
4. Interface and network issue.
5. Failure to utilize the complete capabilities of the Test Systems.

Beyond these five critical issues, the System B has so far supported the mission that it was targeted for. However, it must be noted, that as it was in the case of System A, the System B was chartered to be an institutional resource for the support of a class of missions.

In the absence of more information the cause of the lower scores in the model evaluation as shown in Table 5 are as follows:

- The element (b) is score at 80%. Of this, 10% is because it could not be used for the low rate Real Time task (estimated to be 10% of the total data volume), 10% because of the bit slip problem. At a system level, synchronization is at least a third of the total functionality of the system, hence the drop of 30%.

- Element (e) failed to transfer all the data to the next task in the Data Capture Front-End subsystem, and thus the drop of 30%. At a system level the drop is 40%.

- Element (i) failed to deliver the Real-Time data from the Data Capture Front-End subsystem to the next functional element in System B, hence a drop of 10%. At a system level, the Data Capture Front-End task is at least a third of the total functionality of the system, hence the drop of 30%.

- At a system level, the failure to provide the Real-Time function with the developed prime system derates the capability for elements (a), (c), (d) and (h).

Table 5 Data Capture Front End System Model Evaluation

Function	Element	System	Rank
a) Ingest data correctly	100	90	2.0
b) Synchronize data stream	80	70	4.0
c) Maintain quality statistics	100	90	2.0
d) CRC Decoding/Detecting	100	90	2.0
e) Transfer data from Ingest to detect/decode/correct	70	60	5.0
f) Reed Solomon Decoding/Detecting	100	100	1.0
g) Reed Solomon Correction	100	100	1.0
h) Maintain quality statistics	100	90	2.0
i) Deliver data from detect/decode/correct	90	70	3.4
			2.49

System C – Performance Measurement and Model Evaluation: Unlike Systems A & B, System C was essentially a proof of concept design prototype to demonstrate new technologies in meeting the critical requirements that were levied on System B. However, since it was to demonstrate the proof of concept, the driving factor in the implementation was capability and cost.

Table 6 System C - Real Time Data Transfer Performance – Measured

System Requirements (Mbps)					Prototype (Mbps)	Measured (Mbps)		
Requirement	Sys.	FES	Proc.	Xfer.		Prel.	Initial	Final
Single Svc	150	>350	>400	>400	150	150	167	190
Multi-Svc	150	>350	>400	>400	150	110	131	165
Single Svc w/RT Stat	150	>350	>400	>400	150	75	110	150
Multi-Svc w/RT Stat.	150	>350	>400	>400	150	35	90	130

The model evaluation for System C Front End System is shown in Table 7, From the analyses three major setbacks were attributed and identified as described in the following paragraphs.

Table 7 System C Front End System Model Evaluation

Function	Element	System	Rank
a) Ingest data correctly	100	100	1.0
b) Synchronize data stream	100	100	1.0
c) Maintain quality statistics	100	70	3.1
d) CRC Decoding/Detecting	100	100	1.0
e) Transfer data from Ingest to detect/decode/correct	100	100	1.0
f) Reed Solomon Decoding/Detecting	100	100	1.0
g) Reed Solomon Correction	100	100	1.0
h) Maintain quality statistics	100	70	3.1
i) Deliver data from detect/decode/correct	100	30	6.0
			2.02

- Even though the commercially accepted standard PCI bus was being widely used, there were still many holes in the formalization of the standard and not enough statistical or historical information. This decreased the validity and veracity of the data pertaining to commercially available PCI interface devices, which in turn restricted the data transfer rate from the Front-End system to the Disk storage system to less than 210 Mbps.

- The mid-level workstation platform used to facilitate the data transfer from the Front-End System across the Host Platform PCI bus to the Disk storage system had degraded performance on the initiation of a second channel of information namely the status and monitoring data.

- Incorrect coding of the VHDL and hence the incorrect synthesis of a programmable device caused data corruption of a byte every three seconds when running at 150 Mbps.

CONCLUSIONS

Table 8 Comparisons for A, B & C Front-End Systems

Performance Measures	SYSTEM A FRONT-END		SYSTEM B FRONT-END		SYSTEM C FRONT-END	
	Modeled	Measured	Modeled	Measured	Modeled	Measured
Functionality	*1.62*	4.47	*1.85*	2.25	*1.54*	3.92
Flexibility	*3.85*	3.35	*3.93*	2.25	*1.55*	2.94
Rel. & Maint.	*4.09*	8.95	*4.17*	4.50	*1.48*	1.47
Cost/Schedule	*2.35*	1.12	*1.95*	1.69	*1.67*	0.98

From the tabulated results in Table 8 the following conclusions are drawn:

- In all three systems the Performance Measure that digressed most from the modeled value is Functionality. This may be attributed to the fact that the factors that most affect this measure are

Technology, Complexity and *State-of-the-Art*. If we examine these factors closely, one major contributing factor could be the lack of prototyping. In evaluating the responses from the technical personnel involved with the testing of these systems, this presumption was validated.

- In all three systems there is a common trend in the modeled versus the measured rankings. The exception is the Flexibility measure for System C. This could be attributed to the fact that System A and B were developed to be operational systems, the requirements were evolving and hence they were modified when necessary to fit the capabilities. When the System A or B could not meet the requirements for Flexibility, the requirement was relaxed and the Systems were then within specification. However, for System C, the prototype was built to demonstrate the technology and the system developers had to identify and isolate the loss in Flexibility. One major source of the 'In-Flexibility' was the inability to make full use of the commercially available interface device as advertised to enhance the capability of System C. It is evident that the lack of information is the major cause of the problem. Since, in the absence of information, the only recourse is benchmarking, the result is self-evident, and prototyping would have reduced this risk.

- In two of the three systems there is a common trend in the modeled versus the measured rankings for the Reliability & Maintainability measure. The exception being the measures for System C. The original System A failed to achieve a major functionality, and hence was deemed to be less reliable. However, System A was the earliest of the three systems developed, and hence had the most amount of uncertainty in the technology. The information was collected at different stages in the design cycle for the three systems. In the case of A, the information was collected in the integration and test phase. Whereas in the case of B, the information was gathered only when the problems were fixed and as such the actual problems may already been resolved. However, in the case of C, the prototyping paradigm was used to a greater extent and hence the difference in the Reliability & Maintainability ranking is very much lower. The fact that the measured is less than the modeled is because the test phase did not stress the system, but was used to demonstrate the proof of concept.

REFERENCES

1. Mirchandani, Chandru. Incremental Prototyping, A Model-Based Methodology For Optimal System Design, Masters Thesis, Reliability Engineering, University of Maryland, 2000.

2. Mirchandani, Chandru. Predict the Reliability of a New VLSI Card Based on Limited Information, 6th VLSI Symposium on VLSI Design, 1997.

3. Hoyland, A., Rausand, M. System Reliability Theory, Models and Statistical Methods, John Wiley, 1994.

5. Mosleh, Ali. Elicitation and Use of Expert Opinion in Reliability and Risk Assessment, Advanced Reliability Analyses, Reliability Engineering, University of Maryland, 1998.

5. Modarres, Mohammad. Reliability and Risk Analysis, 1st Edition, Dekker, New York, 1993.