FEASIBILITY STUDY FOR A DISTRIBUTED SYSTEM DESIGN FOR A GASOLINE RATION CHECK ISSUANCE AND RECONCILIATION SYSTEM

Authors

Olken, Frank
Eggers, Susan
Holmes, Harvard
et al.

Publication Date

1979-09-01

Lawrence Berkeley Laboratory

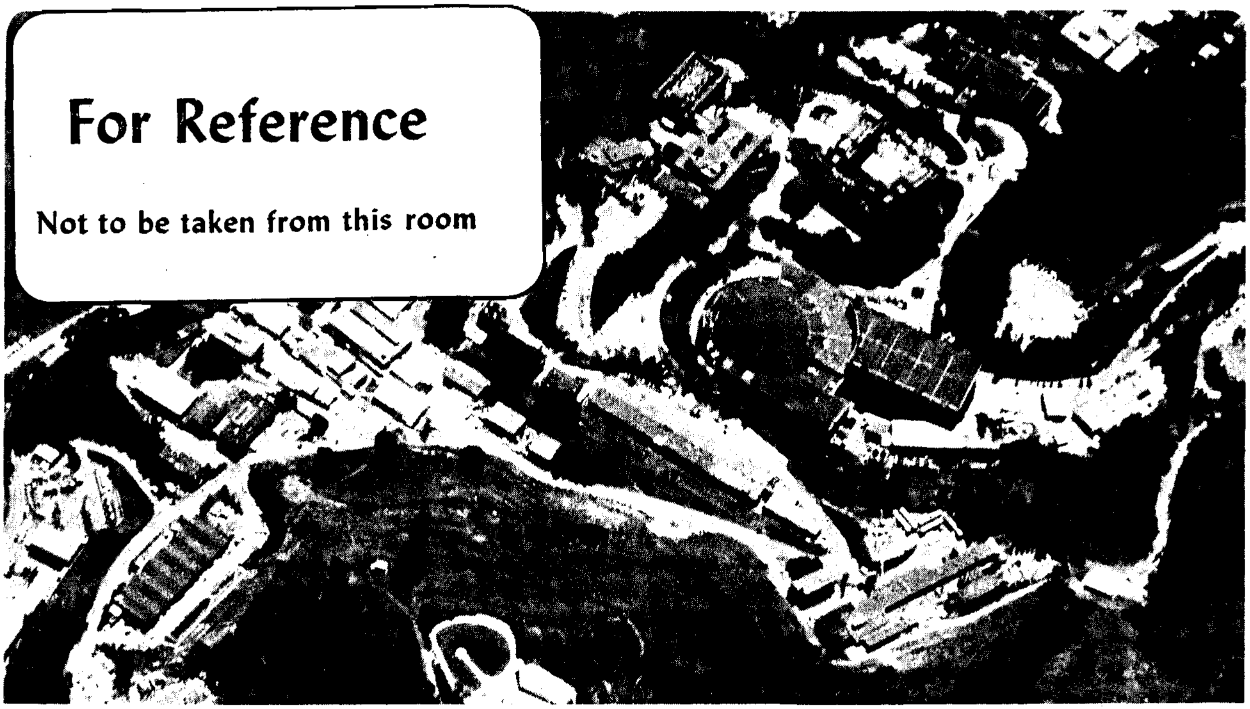
UNIVERSITY OF CALIFORNIA, BERKELEY, CA

Physics, Computer Science & Mathematics Division

FEASIBILITY STUDY FOR A DISTRIBUTED SYSTEM DESIGN FOR A GASOLINE RATION CHECK ISSUANCE AND RECONCILIATION SYSTEM

Frank Olken, Susan Eggers, Harvard Holmes, and Arie Shoshani

September 1979



Prepared for the U.S. Department of Energy under Contract W-7405-ENG-48

DISCLAIMER

LBID-101

FEASIBILITY STUDY FOR A DISTRIBUTED SYSTEM DESIGN FOR A GASOLINE RATION CHECK ISSUANCE AND RECONCILIATION SYSTEM

Frank Olken
Susan Eggers
Harvard Holmes
Arie Shoshani

Computer Science and Applied Mathematics Department
Lawrence Berkeley Laboratory

September, 1979

This work was supported by the U.S. Department of Energy under Contract No. W-7405-ENG-48.

TABLE OF CONTENTS

1. Introduction

This report concerns the issuance, cashing, and reconciliation of ration checks for gasoline rationing. We are primarily concerned with the data processing requirements of a gasoline rationing plan.

This is one of several reports on this subject. A companion report by Oak Ridge National Laboratory is concerned with a centralized batch processing system. This report discusses distributed (multiple CPU) online systems.

The intent of this report is to provide the reader with an understanding of the issues, costs, and feasibility of various distributed online ration check issuance, cashing and reconciliation systems.

We are primarily concerned with systems designed to minimize the inconveniences to the public by gas rationing. Another major objective is to minimize the opportunities for fraud.

We expect that this report will be read by persons trying to decide which type of system to adopt for gasoline ration coupon issuance and reconciliation. If one of the systems discussed here is adopted, we then expect that this report will be used by those who will write an RFP.

The emphasis of the report is on aspects of the systems which differ substantially from centralized batch systems.

1.1. Organization of the report

(1) Introduction

(2) The Ration Plan

(3) Assumptions

(4) Advantages of an Online System

(5) File Organization

(6) Data Acquisition

(7) Data Integration

(8) Ration Check Address File Generation

(9) Ration Check Production

(10) Mailing

(11) Coupon Issuance Points

(12) Registration and Coupon Issuance Points

(13) Reconciliation

(14) Communications costs

(15) Hardware Configurations

(16) Software Requirements

(17) Privacy Considerations

(18) Summary

2. The Rationing Plan

2.1. The Basic Rationing Plan

We are only concerned with ration check issuance, cashing and reconciliation for gasoline rationing.

Ration allotments will be made on a per vehicle basis and will depend on the type of vehicle, its fuel type and weight. A National Vehicle Registration File (NVRF) will be maintained to record this information and the name and address of the vehicle owner. Each quarter, a "ration check" will be mailed to the vehicle's registered owner. The owner will take the ration check to a Coupon Issuance Point (CIP) along with his vehicle registration papers and personal identification. There he will present these documents, endorse the check, and receive ration coupons. The ration coupons will be negotiable. Thus vehicle owners who want more gasoline can buy ration coupons from those who do not need their entire allotment.

For a variety of reasons (change of residence, transfer of ownership of vehicles, errors in registration records), some vehicle owners will not receive their ration checks. Such persons are called "nonrecipients" and are referred to RCIPs (Registration and Coupon Issuance Points) to claim their allotments.

2.2. Time Schedule

It has been proposed that the rationing system would go from standby status to operation in 45 days. Allowing some time to start CIP's, mailing operations, etc., we calculate that four weeks will be allowed in which to process all transactions.

Once gas rationing has begun we assume that the issuance of allotments for subsequent quarters will be staggered, so that each month one third of the population will receive their allotment checks for the next three months. The workload will therefore be spread evenly over each quarter.

2.3. Revisions to Ration Plan

We shall assume that ration checks must be cashed by the vehicle owner and can not be endorsed to second parties. The original Standby Gas Rationing Plan permitted endorsements to second parties - typically gas station operators. The fraud prevention schemes discussed in this report hinge on the fact that all check cashing transactions are verified and recorded immediately via computer terminals. Permitting the endorsement of ration checks to second parties would defeat these fraud countermeasures. We have therefore assumed that the plan would be changed to prohibit such second party checks.

We also assume that one check is mailed for each registered vehicle. There has been some discussion of "netting"[1] operations which would combine checks intended for a single vehicle owner. Such netting is expected to offer a savings of perhaps 20% in mailing costs but would complicate the design and operation of the system. Especially difficult to reconcile are those cases in which the owner has sold some, but not all of his

[1] Cf. R.L. Polk and the Oak Ridge Reports

vehicles. This would require partial cashing of ration checks. Since allotments for each vehicle must be reconciled individually we anticipate little reduction in reconciliation costs. In the interests of simplicity we have discarded netting.

There has been discussion of a proposal to limit the total allocation to a household. Such a rule would require that a form of netting be performed. How this would affect households of unrelated persons is unclear. (Would ration rights become a form of community property among the members of the household?) One version would limit total allocations per family. We have no means of identifying families if the surnames of all family members are not identical. Furthermore such a rule would inevitably discriminate against extended family households. Because we regard such rules as impractical to implement we will not discuss them further.

3. Assumptions

3.1. Numbers of transactions

Each quarter a check will be distributed for each of the 150 million vehicles in the U.S. It is expected that 10% of the vehicle registrations will be in error (due to changes in residence or ownership). Thus, 135 million CIP transactions and 15 million RCIP transactions are expected. Note that we have assumed that RCIPs issue coupons, rather than checks.

In addition, for every RCIP transaction we can expect a corresponding transaction from the state DMV. Such transactions would be reported on tape weekly. They can be processed in background during slack periods.

The reader should be aware that there is reason to believe that the 10% error rate may be far too low an estimate for errors in the initial NVRF. Substantially higher error rates would be extremely difficult to cope with during the proposed startup schedule. If it appears that much higher error rates will occur we would urge that either:

(1) the startup schedule be relaxed

(2) a full scale trial mailing be conducted (with RCIP in operation) over an extended period (3 months) to correct the NVRF during the pre-rationing period.

3.2. Schedule Scenarios

3.2.1. Worst Case Startup Schedule

We assume four weeks are allowed in which to process all transactions. All offices are open 7.5 hours/day, 5 days/week. (This allows 15 minutes to open and 15 minutes to close the office.) Therefore 150 hours are required to process all transactions. Peak traffic is 4 times average. Thus peak traffic rates are 3.6M CIP transactions/hour (1000/second), and .4M RCIP transactions/hour (110/second).

3.2.2. Best Case Startup Schedule

We assume 4 weeks are allowed in which to process all transactions. All offices are open 12 hours/day, 6 days/week, providing 288 hours to process all transactions. Peak traffic is 2 times average. Thus peak traffic rates are 938K CIP transactions/hour (260/second) and 104K RCIP transaction/hour (29/second).

3.2.3. Worst Case Normal Schedule

We assume that 12 weeks are allowed in which to process all transactions. (This assumes staggered issuance of quarterly ration allotments, one third of the allotments issued each month.) All offices are open 7.5 hours/day, 5 days/week. (This allows 15 minutes to open and 15 minutes to close the office.) Therefore 450 hours are required to process all transactions. Peak traffic is 4 times average. Thus peak traffic rates are 1.2M CIP transactions/hour (333/second), and 133K RCIP transactions/hour (37/second).

3.2.4. Best Case Normal Schedule

We assume that 12 weeks are allowed in which to process all transactions. (This assumes staggered issuance of quarterly ration allotments, one third of the allotments issued each month.) All offices are open 12 hours/day, 5 days/week, plus 3 hours on Saturday, providing 756 hours to process all transactions. Peak traffic is 2 times average. Thus peak traffic is 357K CIP transactions/hour (99/second) and 40K RCIP transaction/hour (11/second).

3.2.5. Summary

Transaction Rates
at peak load

Scenario	CIP transactions per hour	CIP transactions per second	RCIP transactions per hour	RCIP transactions per second
Worst Case Startup	3600K	1000	400K	111
Best Case Startup	938K	260	104K	29
Worst Case Normal	1200K	333	133K	37
Best Case Normal	357K	99	40K	11

3.3. Terminal Configuration Scenarios

3.3.1. RCIPs only online

The RCIPs would have alphanumeric CRT terminals which are online to the rationing database. Inquiry of the database would stop certain types of fraud. Online entry of changes to the database offers improved opportunities for catching and correcting errors.

CIPs would be completely manual. Reconciliation would be batch.

3.3.2. RCIP and CIPs both online

The RCIPs would operate as in Section 3.3.1 The CIPs would have calculator type terminals connected to the rationing database. Reconciliation would be done online, thus considerably enhancing opportunities to detect fraud.

3.3.3. Automatic Teller Machines

CIPs would use automatic cash dispensing machines rather than tellers to dispense ration coupons. Advantages include reduced labor costs and extended operating hours. Online reconciliation prevents many types of fraud.

3.4. Human Transaction Processing Times

3.4.1. CIP

We expect that a CIP transaction (exchanging a ration check for coupons) will require an average of 3 minutes. This number comes from the PRC[2] report on CIPs.

[2] Quick Reaction Work Order: Identification of Coupon Issuance Points

3.4.2. RCIP

We expect that an RCIP transaction (entering a change of address, or change of ownership, and obtaining coupons) will require an average of 10 minutes. This figure is based on inquiries of a local California DMV office, where a simple vehicle registration requires 10 minutes.

3.4.3. Transaction Retries

The above transaction times include time to retry 20% of all transactions which we assume will fail because of keying errors. CIP keying errors will be caught by check digit verification algorithms in the concentrators. RCIP errors will only be discovered in the mainframe.

3.5. Uniform Population Distribution

In calculating the number of teller windows required, and in calculating communications costs, we have assumed the population to be uniformly distributed across the country. Additional teller windows may be required to serve sparsely populated areas.

The extent of the error introduced by this assumption is largest for the more modest scenarios.

This assumption was made to simplify calculations.

3.6. Numbers of RCIP and CIP teller windows

The following table was computed by dividing the assumed transaction rates (Cf. Schedule Scenarios, Section 3.2) by the assumed human transaction processing times (Cf. Section 3.4).

Number of teller windows required for peak load

Scenario	CIP	RCIP
Worst Case Startup	180K	66.7K
Best Case Startup	47K	17.3K
Worst Case Normal	60K	22.0K
Best Case Normal	17.85K	6.67K

3.7. Number of teller windows per site

We expect very few teller windows per RCIP or CIP. One reason is the desire to maximize the geographic dispersion of RCIPs and CIPS to minimize travel times and costs for the public. The second reason is that if facilities are shared with other institutions (banks, post offices, DMV offices) we expect to obtain very little space at each site.

Hence we will assume 1.5 CIP teller windows per CIP and 1.0 RCIP teller windows per RCIP. This yields the following estimates for the number of CIPs

---

(106c) Volume 1, page 12, PRC Energy Analysis Corporation Final Report, R-1868, January 1979, Prepared for DOE Contract ET-78-C-01-3361 and RCIPs:

Number of locations
required for peak load

Scenario	CIP	RCIP
Worst Case Startup	120K	66.7K
Best Case Startup	31.3K	17.3K
Worst Case Normal	40K	22.0K
Best Case Normal	11.9K	6.67K

3.8. Number of Processing Sites

Our nominal design assumes the system to be partitioned into 10 sites, each located at the administrative capital of a federal region and serving that region. See Section 14, entitled "Communications".

3.9. Reliability

We are assuming that 100% computer availability is not necessary. Hence manual backup procedures will be employed by CIPs and RCIPs in the event that the automated sytem is unavailable.

However the goal is that system outages will average less than 10 minutes in duration and that they will occur less than once per week.

3.10. Disk Sector Size

We shall assume that disk sector sizes are approximately 1 kilobyte. There are two reasons for this assumption. One is that IBM's transaction processing operating system (ACP) does not support larger sector sizes. The other is a desire to minimize the loading of disk controllers and channels so as to reduce hardware costs. (Cf. Section 15.3.2 for a more detailed explanation.)

3.11. Miscellaneous

(1) There will be multiple CPU's.

(2) There will be communications between all CPU's.

(3) We expect high locality of reference (to vehicle records) by geography, e.g., most people will cash their ration checks near their residence and most moves are within a federal region.

(4) The database(NVRF) will be partitioned geographically by sites, i.e., each site will serve one geographic region.

(5) Systemwide management information system services will run on a summary database (updated nightly) at a single site. This will avoid the complications of a general purpose distributed database management system.

4. Advantages of an Online Distributed System for Gas Rationing

In an on-line distributed system, every CIP will have at least one, and possibly several terminals that will facilitate the dynamic verification of checks and the recording of check cashing transactions. Similarly, RCIPs will have more sophisticated terminals which will allow checking, updating and insertion of NVRF records. The NVRF will be partitioned and reside on multiple (but similar) machines. In this section the advantages of such a system will be described.

An on-line approach provides advantages in the areas of reconciliation and fraud prevention, but does not necessarily imply a distributed implementation of the NVRF. Therefore, we will first describe the advantages of a distributed system, and follow with a discussion of reconciliation and fraud prevention.

4.1. Advantages of a Distributed System

The term "distributed system" implies only that the NVRF will be partitioned on multiple machines. It does not imply that the machines have to be physically distributed at different sites or that there is on-line access to them. Therefore, we emphasize the advantages stemming from physical distribution. However, since the advantages in reconciliation and fraud prevention are the result of the on-line capability, we will assume an on-line distributed system for the following discussion.

4.1.1. Feasibility of Implementation

An on-line system generates a large number of transactions, and requires response time with minimal delay. In the Assumptions chapter (Section 3.2.5) it was estimated that there will be 1000 CIP transactions per second at peak load for the worst case startup scenario. This level of activity generates such a high degree of input/output activity that it cannot be handled by a single computer available today. (The largest IBM system used for an airline reservation system, an IBM model 3033, can handle about 200 transactions per second). The central computer will become the bottleneck to the system. If the NVRF is partitioned into several machines, perhaps on the order of 10 (for the 10 federal regions) each machine could handle 200 transactions per second.

4.1.2. Flexibility of System Configuration and Growth

A distributed system composed of several small machines is more amenable to change than a single large centralized machine. A distributed system allows for development, implmentation, and incremental growth, and lends itself to reconfiguration in case of load shifts.

4.1.3. Reliability

In a distributed system a single machine failure does not cause the failure of the entire system. Also, one can more readily afford to maintain a spare machine to replace the machine that failed until it is repaired, thus reducing down time. A spare machine for a single centralized system is more expensive.

4.1.4. Safeguard Against Catastrophe

A physically distributed system provides inherent protection against catastrophe. If the entire system is located in a single building, then acts of sabotage, inadvertent damage to utilities, or a natural catastrophe could disable the entire system. In a physically distributed system a single failure impacts only a portion of the system.

4.1.5. Local Control and Privacy of Information

Local control and privacy of information issues tend to favor a physically distributed system. Such a system can naturally accommodate local control of files. It can be designed to prevent access to information about an individual's change of residence or similar types of information. In addition, it may be more difficult for unauthorized users to search the entire NVRF. Even if an unauthorized person succeeds in gaining access to local files, more steps are needed to obtain information at remote sites.

4.1.6. Some Communication Cost Savings

The great majority of the communication traffic is local to a federal region. Clearly, CIPs deal primarily with local people. RCIPs perform multiple functions, most of which are local, but relatively few generate inter-site traffic, and some may require a global search on all sites.

Intuitively, it seems that a physically distributed system would minimize communication costs. However, as discussed in the Communications Chapter the bulk of communication costs are between terminals and concentrators, and therefore the cost savings is small.

In addition, there are other costs that grow with the number of sites, such as the cost of buildings, maintenance, and additional spare hardware for reliability. These limiting factors also contribute to the choice of a modest number of sites.

4.2. On-line Reconciliation Advantages

On-line ration check reconciliation offers many advantages over "offline" reconciliation. In an on-line system, checks are compared with the NVRF at the time of the transaction and the transaction is recorded immediately. Similarly, at RCIPs, any coupons issued are recorded immediately. The main advantages are as follows:

4.2.1. Reduction of the need to process the checks.

Offline check processing requires special magnetic ink printing of checks, or a standard font for optical character readers, (OCR's) and the machines required to the read the checks. Such encloding of checks might still be desirable to process returned checks (about 10%).

4.2.2. Reduction of errors

Errors can be introduced if checks are damaged, mutilated or lost before they are reconciled. Resolving such errors may prove difficult and expensive. In an on-line system, the information must match both the check and the file or the transaction will not be authorized.

4.2.3. Reduction of Enforcement Costs

On-line reconciliation eliminates the need to trace people who received coupons incorrectly. For example, a person may receive his check late, and be given coupons at an RCIP. He may subsequently receive his check in the mail, and cash it. This duplicate issuance of coupons will be detected only in the reconciliation phase and will require additional steps (and cost) to remedy the situation.

4.3. Fraud Prevention

If fraud is excessive, it could render the gas rationing system useless. The fraudulent addition of gasoline coupons to the market can negate the effect of the rationing by creating more coupons than available gas. Failing to design the system with fraud prevention in mind creates undue opportunities for special interest or criminal elements to break the system through fraudulent use. Even if no organized fraud exists, the system might fail unless enforcement is successful.

In a "offline" system, where information is not immediately available, there is little choice but to believe people who claim they did not receive checks[3]. It leaves an opening for fraudulent acts that can only be remedied after reconciliation. In addition, the probability of finding an offender who used forged documents is small. An on-line system can prevent most of the fraud possible in a gas rationing system. It offers not only a great psychological deterrent, but also cost advantages in terms of tracing and prosecuting offenders. In the following sections, we identify the possible types of fraud and show how they can be prevented with an on-line system. We also give an estimate of the cost reductions achieved by fraud prevention.

4.3.1. Types of Fraud

Fraud can be classified into two categories, which we will label soft fraud and hard fraud. Soft fraud involves a one-time act and is easy to commit and difficult (or expensive) to prosecute. For example, consider again the case of the person who claims to an RCIP teller that he has not been mailed his check, is issued coupons, and upon receiving a check in the mail, cashes it. It would be very difficult to prove that this person acted maliciously. Fraud of this type is particularly dangerous in that if not fully and successfully prosecuted it could grow to excessive levels.

Hard fraud involves the use of forged documents or stolen checks and is more expensive to prosecute. It may be difficult to locate the people involved because they use false documents. Hard fraud, using forged documents might lead to large scale fraud by organized crime. For example, it is possible to cash 1,000 forged checks in only two or three weeks -- worth $150,000 at $1.00 per coupon, 50 coupons per check -- provided you had the appropriate false driver's licenses and registrations. The cashing period involved is less than the time needed for the offline reconciliation process to reveal the fraud. An additional danger of large scale fraud is that relatively small number of people can cause the system to fail, by creating more coupons than available gasoline.

---

[3]The alternative, to delay issuance of the new check, while issuing a stop order on the earlier check is probably not acceptable to the public.

The distinction between soft fraud and hard fraud is useful in estimating fraud rates, as will be discussed later.

4.3.2. Possible Fraud at CIPs

(1) Cashing a check after coupons have been received at an RCIP for the same vehicle. An online CIP will detect this easily, given that vehicle records are marked when coupons are issued for the vehicle.

(2) Cashing stolen checks. In an online CIP the teller will be required to enter both the check number from the ration check and a portion of the VIN from the vehicle registration (the VIN is not printed on the check, to force the checking of vehicle registrations). The system will not authorize the transaction unless both identifying data elements belong to the same record. This scheme also prevents tellers from cooperating in cashing stolen checks.

(3) Stealing checks and forging car registrations. This will not be possible with the scheme above, wherein VINs are not printed on the check. In order to cash a stolen check it would be necessary to locate a car and gets its VIN for the forged car registration or steal the corresponding registration card.

(4) Cashing forged checks, with a corresponding forged car registration and drivers license. Even if the forged check number is valid, there is still the problem of matching it to a VIN. An on-line system prevents this by the method described above.

(5) Cashing forged checks. This case is similar to cashing stolen checks (case 2). The on-line method above will prevent it, even with cooperating tellers.

(6) Tellers failing to record transactions. A teller may cooperate with a recipient, and dispense coupons for a forged check, simply by not recording the transaction. To prevent the non-recording of transactions a two part logging system will be implemented. In the first part the teller is required to write down the number of coupons issued per transaction on a tally sheet. In the second part he is required to enter this same number into the database. At the end of each day the logs can be compared and discrepancies investigated. This procedure will also ensure that transactions which were inadvertently not recorded (or not recorded due to machine malfunctions) will be detected.

4.3.3. Possible fraud at RCIPs

Fraud at RCIPs is similar in many ways to that at CIPs. However, RCIPs perform the additional functions of creating new NVRF records or modifying existing ones, creating possibilities for additional types of fraud.

(1) Claiming non-receipt of check after the check was cashed. This is easily detected in an on-line RCIP. If the claim is still made that the check has not been cashed, coupons can be issued, identifying information taken, and the car physically inspected. The case can then be turned over for signature verification, etc.

(2) Tellers not checking car registrations and other identification properly. This is handled by requesting information taken from the required

documents, before authorizing the transaction.

(3) Claiming non-receipt with forged car registration and driver license. This is handled the same as in CIPs.

(4) Tellers entering false new records, presumably for new cars. In an online system new VINS will be checked for their correctness (e.g., VINA program developed by R.L. Polk). Dealers can be requested to report new car shipments; junked cars will be handled similarly.

4.3.4. Additional Fraud Introduced by an Online System

(1) RCIP Line tapping. By tapping into the lines connected to an RCIP a clever interloper could add new records to the database or modify the existing records. The possibilities for tampering with the database in this way justify the use of encryption techniques.

(2) CIP Line tapping. For CIPs it would be possible for a tapper to simulate a computer response to a CIP terminal and authorize the cashing of a forged check. However, this problem is solved by simply printing a random number on the check, and having the computer send it back for verification. Unless this number matches the one on the check, it is not authorized.

4.3.5. Cost Savings Estimates

In order to get some idea of fraud rates and recovery costs we contacted several federal agencies informally and attempted to find situations similar to those anticipated in a gas rationing system.

There is a cost figure associated with the investigation and recovery of fraud, which depends on the type of fraud. A typical cost figure was quoted by the Secret Service Forgery Division (SSFD) as an average of $274 per forged treasury check (checks stolen and signed fradulently). About 30% of the cases go to court, which involves an additional cost (amount unknown, probably on the order of a few thousand dollars). Other types of fraud, such as embezzlement may have a much higher enforcement cost figure (the FBI will not investigate an embezzlement of less than $5,000).

It is typical of many agencies not to investigate fraud under a certain amount due to the cost involved. It is worth noting here that such practical considerations are not very appropriate for gas rationing, as one cannot permit the existence of more coupons than available gas.

Obviously, the level of fraud experienced in any system depends on the opportunities available, and therefore may vary widely. The SSFD quoted a rate of 0.016% for forged treasury checks (112,000 out of 700 million) for Fiscal Year 1979. In the report by the Federal Advisory Committee on False Identification (CFI) it was stated that 0.65% of the checks in the United States failed to clear, but only 0.1% proved to be counterfeit.

The Social Security System (SSA) issues approximately 32 million checks per month (predominently retirement checks) and receives approximately 100,000 complaints of non-receipt. About 60,000 of those were cleared up by checking files and telephone calls. The remaining 40,000 required signature verification. Of those about 60%--24,000 were cashed improperly. If we translate these figures to percentages, we find that about 0.3% (100,000 out of

32,000.000) of the people complained of non-receipt incorrectly. This is equivalent to the soft fraud rate since it requires a relatively small cost to recover. We assume here that losses suffered as a result of soft fraud can be recovered by notifying the offender and threatening action. About 0.075% (24,000 out of 32,000.000) of the checks were actually cashed improperly. This is equivalent to hard fraud in terms of recovery costs, since it will require extensive tracing procedures and perhaps court costs.

Based on the above figures we will assume the following:

```
Soft Fraud (Per Quarter)
     Rate:   0.3%           (SSA estimate)
     Cost:   $50 - $100     (to notify offenders and recover)
     Total:  22.5M (150M x 50  x 0.3%) to
             45.0M (150M x 100 x 0.3%)

Hard Fraud (Per Quarter)
     Rate:   0.016%          (SSFD estimate) to
             0.1%            (CFI estimate)
     Cost:   $274            (cost recovery) plus
             $1000 x 30%     (estimated court cost x 30%
                             of cases) = $574

     Total:  13.7M (150M x 0.016% x 574) to
             86.1M (150M x 0.1% x 574)

 Total cost per quarter:
             36.2M to 131.1M

 Total cost per year:
            144.8M to 524.4M
```

5. NVRF File Organization

5.1. NVRF File Structure

All NVRF data files will be random access, but readable sequentially. Records will be stored in 1K disk sectors.[4] A hash on one of the data elements will produce the disk and sector number on which a particular NVRF record resides. If the records within the sectors are not sorted by the key, a sequential search of the sector will be necessary to obtain the record. If the records are sorted, the sort must be maintained whenever the file is updated.

5.2. Record Structure

The NVRF records will be fixed length, with continuation records only if needed for multi-registrations and temporary duplicate information from RCIP activity. The fixed length scheme was chosen both for simplicity and because the number of continuation records is anticipated to be a small proportion of the database.

Two record format schemes were considered. In one, hereafter known as an integrated NVRF, all NVRF data elements are stored in the same record and are accessed via the hashing and indexing methods suggested in Sections 5.5.4 and 5.5.5.

In the other, hereafter known as a segregated NVRF, those data elements used for CIP check reconciliation[5] are housed in a file separate from the vehicle and owner information. Since the check reconciliation file is recreated each quarter it can be accessed via a hash on a changing check number.

The segregated NVRF has several disadvantages. It would cost the RCIPs three extra seeks per transaction to obtain the check cashing portion, bringing the total number of seeks to 12.6, excluding the 1.8 seeks needed for possible sector overflow and address-zip code matching[6]. It would also require, for each vehicle record, generating and storing a pointer to its check reconciliation record and storing duplicate copies of the VIN or license plate number or both, one in each portion of an NVRF record. In addition it requires extra processing for address file generation and check number assigning. Given these disadvantages a segregated NVRF would only be used if it is both necessary to generate completely new check numbers each quarter <u>and</u> eliminate the 1.2 seeks/transaction needed to access the file of NVRF pointers during CIP activity.

Most of the discussion and calculations in this report have been based on an integrated NVRF.

---

[4] 1K was chosen both for compatibility with the 1055B physical record size of IBM's ACP operating system and for hardware considerations. See Section 15.3.3 for a more detailed explanation.

[5] CIP/RCIP identification, teller identification, date of transaction, check number, allotment, random response digits, special RCIP procedure flag.

[6] See seek calculations, Section 5.6

5.3. Data elements

For processing simplicity, data elements within the records will have fixed positions and lengths.

5.3.1. Mandatory data elements

The following data elements are considered to be essential to the implementation of a distributed RCIR system.

(1) Vehicle identification number (VIN)
This data element can be the basis for the index used by the RCIPs. It is currently composed of seven alphanumeric characters indicating some combination of the make, model, etc. of the vehicle and a six digit production number. Its length is thirteen bytes at maximum; ten are possible if six bit packing is used. Seventeen bytes has been proposed for a new VIN standard. The expansion is in the alphanumeric vehicle description section.[7] Both thirteen and seventeen have been used for space calculations in this report. There is uncertainty as to the reliability of this data element.[8]

(2) License plate number.
May have to be used as the basis for the RCIP index if the VIN is unreliable. Seven bytes, alphanumeric.

(3) Encoded make of vehicle.
Two bytes.

(4) Encoded model of vehicle.
Two bytes.

(5) Model year.
Needed to make the thirteen byte VIN unique. The two lower order digits are stored. Two bytes if stored as characters; one if binary.

(6) Fuel type.
Regular, premium, unleaded, diesel, etc. One byte.

(7) Vehicle category.
Truck, automobile, etc. One byte.

(8) Weight in thousands of pounds.
Two bytes, binary.

(9) State of registration.
For financial reasons, many people register their vehicles in states which are not their states of primary residence. The state of registration will determine in which regional data base an NVRF record will reside. Two bytes, alpha, consisting of the postal abbreviations.

---

[7] Part 571: Federal Motor Vehicle Safety Standards (Vehicle Identification Number), Chapter 5: National Highway Traffic Safety Administration, Department of Transportation, Federal Register, Vol. 44, No. 57, March 22, 1979, p.17498.

[8] Booz Allen & Hamilton, Inc., National Vehicle Registration File Alternative Analysis, Draft Report, July 9, 1979, Exhibit V.

Items (1) through (9) identify the vehicle.

Items (3) through (9) may be used in the algorithm to determine the quarterly gasoline allotment.

(10) Name of owner
There is no pre-defined distinction between the first and last name. Therefore the address file will be sorted on the first digit of the first name. Mailings will be done alphabetically, and in equal size portions. Thirty bytes, alphanumeric.

(11) Street address.
Includes the street number and name, street category, rural route number, apartment number. Thirty bytes, alphanumeric.

(12) City.
Fifteen bytes, alpha. This is one greater than the postal standards.

(13) State.
The standard postal abbreviations will be used. Two bytes, alpha.

(14) Zip code.
Five bytes, numeric.

Items (10) through (14) identifity the owner and are used for mailing the ration checks.

(15) Check number.
Nine digits for a check number, one digit for the quarter, two to four which are random, and two to three for check digits. The minimum number of digits will fit into six bytes in binary and fourteen bytes in character form. The number of bytes for the maximum digit size in both storage forms is eight and seventeen, respectively. The former figures have been used for space calculations in this chapter. This data element is the basis for the CIP hash function.

(16) Date the check is cashed.
Two bytes if stored in binary form as a Julian date; six bytes if stored as month-date-year in characters.

(17) CIP/RCIP identification.
A number which uniquely identifies each CIP and RCIP teller window. Two bytes if the total number of both can be kept below 64K, four bytes if not. Calculations based on this data element have been done for both cases.

(18) Teller.
A number assigned in sequential order to each employee of a CIP or an RCIP. Teller number would be useful in pinning down the source of fraud in the check-coupon exchange. Two bytes, binary.

(19) Quarterly gasoline allotment.
The number of gallons or coupons, whichever is easier from a human engineering point of view. One byte, binary.

(20) Quarterly cashing record.
A check cashing history for the past four quarters. One byte, with one bit per quarter. A quarter's bit is zeroed when the address file for that

quarter is produced and set when the quarter's check is cashed.

(21) Random response digits.
A two digit number (randomly generated by the address file program) which is both printed on the ration check and returned to the tellers to indicate a check number-VIN or a check number-license plate number match. Its inclusion in the CIP/RCIP transaction scenario prevents a line-tapper from simulating the CPU. One byte, binary.

(22) Special RCIP procedures.
An indication of some RCIP activity which designates an irregularity in the check-coupon exchange. Examples are a check number-VIN or check number-license plate number mismatch, missing ration checks, new and traded cars, duplicate quarterly check cashing for a vehicle, and the absence of a record for a vehicle in the NVRF. In addition, there is a value which indicates that any of the forementioned irregularities have been reconciled by the state DMV. One byte, binary.

Items (15) through (22) identify the check cashing transaction.

(23) Pointer to a continuation record.
A single byte (binary) indicating the disk and four bytes (binary) for the sector number.

5.3.2. Optional data element

This data element is considered to be of secondary benefit to the processing of NVRF data or to the detection of fraud. It could be included with no serious ramifications to the space calculations.

(1) Time of the check cashing.
Would aid in the detection of fraud in the case in which one teller had signed onto the CIP or RCIP systems as another teller. Two bytes, binary.

5.3.3. Rejected data elements

The following data elements were not considered of sufficient benefit to the processing of data or detection of fraud to be included in the NVRF.

(1) Body style and transmission type.
Although both may be determinants of gas usage, they are probably of too fine a distinction to be included in any allotment algorithm.

(2) Registration number.
This information is on the vehicle registration document which must be presented in order to cash in the ration checks. It is not necessary to store it in the NVRF record as well; the license plate number is sufficient for proper identification.

(3) Mailing address, city, state and zip.
An alternate mailing address for the ration checks. This data element would add another 50 bytes to each NVRF record, enough to alter most space calculations. In addition it would be used relatively infrequently, e.g., for those students who had cars registered in the state of their parents residence. For those few situations the ration checks can

be forwarded.[9]

5.3.4. Dictionary for textual data elements

The use of a regional dictionary for textual data elements, particularly those whose values are used repeatedly, such as city, should be considered. The entry in the NVRF record would then be a two-byte pointer to the dictionary for the appropriate text. This scheme would result in smaller NVRF records. If the dictionary were disk resident, it would require an extra seek for those data elements whose text is stored for both RCIP activity[10] and address file generation[11]. Alternatively, only the most commonly used text could be encoded and stored in main memory. However, this scheme would necessitate the use of variable length NVRF records.

Decreasing the size of the NVRF record allows more of them to fit into the 1 kilobyte hash buckets, and decreases the size of the database. Once a bucket is located, it is searched (in core) for the appropriate record. If the number of records per bucket makes a sequential search too unwieldy, two alternatives are available. The bucket size could be reduced, thereby necessitatng the use of a greater number of overflow pages. Alternatively, the records could be sorted within the buckets; in this case each record insertion or deletion would probably require a repacking of the bucket.

5.4. Data Base Size

Each integrated NVRF record will be between 134 and 157 bytes in length, using only the mandatory data elements. Seven 146 byte records will fit into the 1K sectors; that figure has been used for all remaining calculations. The total data base size will be 22 gigabytes (GB). Leaving 25 percent of each sector free for expansion brings the figure to 27.5GB.

5.5. Retrieval Methods

Three methods of retrieving information were considered. They are presented below, listing their advantages and disadvantages.

5.5.1. Hashing

Hashing is the computation of a function on a record's key which produces the location of that record. A desirable hashing algorithm will produce a uniform distribution of the records throughout the file.[12]

Advantages:

---

[9] **The Polk report concurs.**

[10] during working hours, if online RCIPs; non-working hours, if offline.

[11] non-working hours, both online and offline RCIPs.

[12] For a discussion on hashing, cf. Donald E. Knuth, The Art of Computer Programming, Vol 3:Sorting and Searching, Addison-Wesley Publishing Co., 1973, beginning p. 506.

(1) No seeks are necessary for the indexing operation. This is the most important advantage, since the system will be I/O bound.[13]

(2) No storage is needed for indices.

(3) The transactions do not involve range searches. Therefore the disadvantage of only exact match searching inherent in hashing does not apply.

Disadvantages:

(1) A portion of each bucket (at least 25%) will have to be kept empty during the initial database building to allow for expansion. This will add 5.5 gigabytes to the size of the database, making the total 27.5GB.

(2) Overflow sectors will have to be used when the original hash buckets have been filled. This will add another seek and additional search time within the overflow bucket to the transaction. Since the data base will grow by 10 percent each year from new cars, all buckets would be filled within two years. A complete data base reorganization might be considered at this time.

5.5.2. Clustered B-tree

A B-tree is a multi-way tree whose nodes are always balanced during file updating. The tree is clustered if the records are sorted by the key. Therefore the index pages can point to sectors of records rather than individual records.[14] Each entry in a B-tree for the NVRF consists of a key, a 4 byte sector number and a single byte disk indicator.

Advantages:

(1) Entries in the index pages can point to sectors of records rather than individual records, thereby greatly reducing the size of the tree.

(2) No additional overflow mechanism is necesary because page splitting preserves the B-tree structure.

(3) Records are sorted by the key which allows a binary search within the page.

Disadvantages:

(1) Because of the large number of CIP transactions per second, each level of a national B-tree would receive approximately 99 to 1000 seeks/second, depending on the scheduling scenario.[15] Regional[16] B-trees would have a

[13]Cf. 5.5.2 and 5.5.3, Disadvantage 1.

[14]For a further discussion on B-trees, see Douglas Comer, "The Ubiquitous B-Tree", Computing Surveys, Vol. 11, No. 2, June, 1979. The article contains an extensive bibliography.

[15]Cf. 3.2 for a description of the schedule scenarios.

[16]To avoid bottlenecking at each level of the tree, the tree can be divided into ten trees, one for each regional center. The number of seeks per level per second then drops by a factor of ten. Dividing an index by regions is consistent with its probable use. In general a person will frequent an RCIP

little more than one tenth the activity. Even under this circumstance only the best case normal schedule functions without a bottleneck. Therefore either the whole tree or its root section would have to be kept in core. In the latter case the index pages could be scattered among several disks, thereby reducing the congestion. For RCIP activity a regional tree is sufficient for all scenarios.

(2) Additional seeks are required for using a disk-resident clustered B-tree. Assuming 146 byte records and 1KB sectors, a nationally based tree using either the thirteen or seventeen byte VIN would require five seeks, and using the seven byte license plate marker, four. For regional trees both keys require four seeks. One seek should be subtracted for each level of the tree which is kept in core.

(3) Additional storage is required to house the clustered B-tree. Storage estimates depend on the length of both the record and the key, and the sector and database size. Assuming 146B records, 1KB sectors and a 150M record database, 393MB of storage would be needed for a thirteen byte VIN-based tree and 489MB for a seventeen. 262MB would be required if the key were the license plate number. (This is not a significant problem.)

(4) When the record pages are full, the B-tree must be updated and the pages split. To avoid continuous splitting, room should be left in the pages for additions, just as in hashing.

5.5.3. Unclustered B-tree

Unlike a clustered B-tree, the records are not sorted on the key.[17] Each entry consists of a key, a four byte sector number and a single byte disk indicator.

Advantages:

(1) Records are stored sequentially. Therefore the need for leaving extra space in the database pages and page splitting in the database are eliminated.

Disadvantages:

(1) There is a bottleneck at each level of the tree, identical to that in the clustered B-tree.[18] Since the unclustered tree is larger than the clustered version, the in-core storage would be a greater overhead and more disk dispersion of the index pages would be necessary.

---

in the region of his/her vehicle registration. If not (the person has moved or lives near a regional border), the former or other region of registration is always known. Rarely will the entire data base have to be searched to obtain a vehicle record. If the region of registration is other than that of the RCIP, the teller will merely enter the region number along with the VIN.

[17] Cf. the reference for clustered B-trees, 5.5.3.

[18] Cf. 5.5.2 Disadvantage (1).

(2) Additional seeks are needed to obtain the NVRF record. If the key is either the VIN or the license plate number, five extra seeks are needed for a national index and four for regional indexes.

(3) Extra storage is needed to house the index. Storage estimates depend on the byte length of the key chosen, as well as the sector size and the size of the data base, the latter of which are fixed at 1KB and 150M records, respectively. 1.86GB, 2.7GB and 3.4GB are required for the index depending on whether the license plate number, the thirteen byte or the seventeen byte VIN is used as the key. Because the records are not clustered, an index page entry is needed for each record. Therefore the storage for the index is greater than for a clustered B-tree.

(4) Also because of the lack of clustering, the records are unsorted in each sector, necessitating a sequential search for a record.

5.5.4. Recommended Indexing for CIP usage

CIPs are expected to have terminals which handle only numerics, and therefore the index to the database for CIP usage must be based on check numbers. Since 90 percent of all transactions will occur at CIPs, resulting in a large number of seeks per second, the priority for CIP processing of the NVRF records is access efficiency. Therefore the retrieval scheme which minimizes the number of seeks was chosen, i.e., hashing.

At least three schemes are reasonable, two for an integrated NVRF, one for a segregated NVRF[19].

(1) Hash to the NVRF Record (Integrated)

Since the placement of the NVRF records is determined by the hashing algorithm, it is imperative that a portion of the check number remain constant. The hashing algorithm would therefore include neither the current quarter digit nor possibly the random digits. The address file generator[20] could assign the check numbers for the first quarter and merely alter the variable portion thereafter. The number of seeks needed for the CIP transaction is: 2 for the NVRF record read and write and 1 for the write required for disk backup, making a total of 3 seeks per transaction, assuming no sector overflows. Since NVRF record positions are determined by a hash on the check number, RCIP access to the file would utilize a secondary index based on the VIN or license plate number. Such a scheme is described in 5.5.5.

(2) Hash to a file of NVRF record pointers. If it is necessary to generate completely new check numbers for vehicle records each quarter, the placement of the records in the file can be determined by a hash on another data element (the VIN or the license plate number), and access via the check number can be implemented by hashing to an index file. The index file will contain a series of check numbers and pointers to the corresponding NVRF records. Each pointer will consist of a one byte disk indicator and a four byte sector number. The file itself will require

---

[19]Integrated and segregated NVRF schemes are discussed in Section 5.2.

[20]Cf. "Ration Check Address File Module".

either 1.65GB or 2.85GB, depending on the storage form of the key.[21] The number of seeks needed for the CIP transaction is: 1 for the NVRF index file read, 2 for the NVRF record read and write, and 1 for the write required for disk backup, making a total of 4 seeks per transaction, assuming no sector overflow. Each quarter the address file generator would assign new check numbers to the vehicles and would update the NVRF index.

(3) Hash to the Check Reconciliation Record (Segregated). If it is necessary to keep CIP seeks/transaction to an absolute minimum and to generate completely new check numbers each quarter, the check reconciliation data elements can be physically separated from the others and accessed by hashing on the check number. The number of seeks needed for a CIP transaction is: 2 for the check reconciliation record read and write and 1 for the write required for disk backup, bringing the total to 3 seeks per transaction, assuming no sector overflow. Each quarter the address file generator would recreate the check reconciliation file, assigning new check numbers to the vehicles, and updating each vehicle record's pointer to its check reconciliation record.

5.5.5. Recommended Indexing for RCIP usage

RCIP activity must depend on a data element other than the check number to access the data base. Since RCIP terminals are alphanumeric, either the VIN or the license plate number can be used.[22]

Three schemes are listed, each corresponding to one of the CIP retrieval methods.

(1) If NVRF records are organized according to a hash function based on the check number,[23] then RCIPs would access the records by hashing (on either the VIN or license plate number) into an index of NVRF record pointers. Each entry in the index file contains one of the keys and a pointer to its corresponding NVRF record. The pointers consist of a one byte disk indicator and a four byte sector number. The total size of the file would be 1.8GB, 2.7GB or 3.3GB, depending on which data element was chosen as the key (license plate number, either of the storage forms for the VIN). The number of seeks needed for a RCIP transaction is: 1.2 for the NVRF index file read, 3.2 for the NVRF record read and write[24] (including a twenty percent teller entry retry for the reads), 3.2 for an expected read and write on the continuation record,[24] and 2 for the writes required for duplicate files, making a total of 9.6 seeks per

---

[21]**Cf. 5.3.1 (15) for a description of the check number data element.**

[22]The accuracy of the VIN is in question. Booz Allen & Hamilton, op. cit., Exhibit V. But license plate numbers are issued up to six weeks after the purchase of a new car.

[23]Cf 5.5.4 (1).

[24]The disk sector must be reread because of the length of time (ten minutes) required by the teller to process the transaction precludes storing the sector in main memory.

transaction, assuming no sector overflow, and excluding the 1.2 seeks needed for a possible address-zip code match. If RCIPs are offline, new vehicle registrations would receive check numbers when the batch update of the NVRF was done. If online, they would be assigned during RCIP transaction processing.

(2) If a vehicle's check number changes each quarter, and the database is consequently organized according to the VIN or the license plate number,[25] access to each NVRF record can be accomplished by a hash on one of the latter two data elements. The number of seeks needed for a RCIP transaction is: 3.2 for the NVRF read and write,[26] assuming a twenty percent teller entry retry for both reads, 3.2 for the read and write on the expected continuation record,[26] 2 for the NVRF record duplication, making a total of 8.4 seeks per transaction, excluding possible sector overflow and 1.2 seeks for address-zip code matching.

(3) If the check reconciliation portion of the NVRF is housed in a file separate from the vehicle registration portion, the vehicle registration portion can be organized according to the VIN or license plate number. Access to it will be accomplished by hashing on one of those data elements; access to the check reconciliation portion will be by a second pointer. The number of seeks needed for a RCIP transaction is: 3.2 for the vehicle registration portion read and write,[26] assuming a twenty percent teller entry retry, for all three reads, 3.2 for the check reconciliation read and write,[26] 3.2 for the expected read and write on the continuation record,[25] 3 for all duplication, making a total of 12.6 seeks per transaction, excluding possible sector overflow and 1.2 seeks for address-zip code matching.

For a RCIP transaction, new transactions would be handled in the same manner as in the hashing scheme in (1) above.

---

[25] Cf. 5.5.4 (2).

[26] The disk sector must be reread because of the length of time (ten minutes) required by the teller to process the transaction precludes storing the sector in main memory.

6. Data Acquisition Module

6.1. Generic Description

The Data Acquisition Module provides at least the initial data sources for the NVRF. Inputs are the states' DMV files, augmented where possible by other sources such as the title files used by some states to maintain individual records. This module is defined to consist of one or more processing centers equipped to receive, account for, and ensure completeness of the data sets. This could be a continuing operaton, as the states send in updates on a regular basis; or, it could be a one-time function, with updates being generated entirely from other modules in the system. The outputs from this module include the complete set of vehicle registration files for building the NVRF and perhaps some feedback to the state DMVs to assist them in correcting their files.

-- Inputs

State DMV files
Auxilliary information from other sources
Updates to state DMV files

-- Outputs

Complete data set for 50 states and D.C. to NVRF

— Function

Receive, account for, and manage data input from states, and provide complete data set in machine readable form for the NVRF processing module. This may require data entry for states which cannot provide machine readable inputs.

6.2. Data Elements

The chapter entitled "File Organization" discusses which data elements are to be acquired.

6.3. Frequency

We are of the opinion that while in standby mode periodic updates from state DMV's should be processed. This will exercise portions of the software (for updating the NVRF) thus providing some assurance of their correct operation.

6.4. Acquisition from non-automated states

The online RCIP system could be used to capture data from states which are not presently computerized.

6.5. Costs

Acquisition costs are similar for online, batch, centralized, or distributed systems. Thus we have not generated independent estimates. The reader should consult the Oak Ridge report on centralized batch systems.

7. Data Integration/NVRF Module

7.1. Generic Description

The Data Integration module creates and maintains the National Vehicle Registraton File (NVRF) using the data sets provided by the Data Acquistion module. Validation and correction of the records takes place in this module, using such techniques as ZIP code matching, VIN validation, duuplicate detection and resolution, and other error correction techniques. This module is responsible for integrating all sources of information for maintaining the highest quality NVRF possible. Correction files can be provided to the state DMVs, if those organizations desire to use these files for updating their records.

-- Inputs

Data sets from the Data Acquistion module
Updates and corrections from Data Acquisition, RCIP, and Reconciliation modules

-- Outputs

The most current NVRF to the Ration Check Address File Module
Correction files to the states

-- Function

Maintain and validate the NVRF for the production of address files for Ration Check Address File module.

7.2. Proposal

We expect to integrate the RCIP functions with data integration. Thus RCIP transactions would be immediately checked for duplicate or inconsistent VIN's. Also we would look up the city name in an in-core table (to check spelling and to locate the hash table of street names and zip codes for that city). The street name would be hashed to find a table (on disk) giving the zip codes for various street numbers. Unknown addresses and other errors would generate error messages. Errors which could not be resolved by the clerk would be flagged in the database and written to a special error log for further investigation.

Updates from DMV's will be processed (from tape) in a manner similar to RCIP transactions during slack periods. Some RCIP transactions (change of residence, transfer of vehicle ownership) will generate corresponding DMV transactions. Inconsistencies will be written to an error log for further investigation by DOE or DMV personnel.

Unresolved errors which affect the zip code would cause the ration check to be issued as part of a segregated mailing.

7.3. Initial Creation

We do not believe that it will be possible to create the NVRF by simply treating all the DMV records as updates and processing them using random access the NVRF. Too many seeks would be required. Instead it will be necessary to:

(1) Sort the input files by address.

(2) Run zip coding programs

(3) Sort by VIN

(4) Remove duplicates

(5) Assign a check number to each VIN.[27]

(6) Build a secondary index based on VIN's or check numbers.

(7) Sort by hash code of invariant portion of check number. (May be unecessary if hashcode is simply the check number.)

(8) Store vehicle registration records into the NVRF according to the hash code of check number or VIN.

The above assumes that we are using an integrated NVRF record consisting of a vehicle registration and check reconciliation portions, stored according to a hash code on the invariant portion of the check number assigned to each VIN or the VIN itself[28].

If we employ a segregated NVRF in which vehicle registration records are stored separately according to a hash code calculated from the VIN, then the creation entails the following:

(1) Sort input file by address.

(2) Run zip coding programs.

(3) Sort registration records on a key composed of the VIN hash code, followed by the actual VIN.

(4) Remove duplicate registration records (for same VIN).

(5) Store vehicle registration records in NVRF according to VIN hash code.

The reader will note that the sorting involved in each proposal will assure us that only sequential file references will be needed. Thus we will avoid numerous time consuming disk seeks.

7.4. Initial Duplicate Removal

[27]The mapping from VIN to invariant portion of check number is not algorithmic because an algorithm would generate too long a check number to be conveniently keyed in at a CIP terminal.

[28]See the chapter "File Organization" for the rationale for such a file structure.

7.4.1. Using a mini-NVRF

If all the files are at one site removing duplicates from a NVRF sorted by VIN is straightforward. If the number of duplicates anticipated is small, and the final NVRF will not be sorted by VIN (or VIN hashcode), then it may be desirable to extract a mini-NVRF consisting of:

(1) The VIN (including the year)

(2) A unique record identfier

(3) A transaction date to resolve which record should be kept

The mini-NVRF would then be sorted. Duplicates would be detected and a file of records which are to be removed from the NVRF would be generated. This small file would be sorted in the final sort order of the NVRF and passed against the sorted NVRF as it is being loaded onto disk. This method should be more efficient, since it avoids an extra sort on the entire NVRF.

7.4.2. Multi-site duplicate removal

If the NVRF is physically distributed across multiple sites (e.g. 10 federal regions) or at a single site, but on multiple machines (each serving a different geographic region), it may be undesirable to attempt to construct a monolithic sorted NVRF.

In such case we would propose the following:

Each site creates a sorted mini-NVRF with duplicates removed, thereby saving a list of duplicates to be purged from the real NVRF. It is possible to detect duplicates across sites in either of two ways:

(1) the mini-NVRF's are all sent to one site where they are merged together, duplicates detected, and lists of duplicates to be purged are prepared for each site.

(2) the mini-NVRF's are split (on the VIN) into n pieces (where n is the number of sites). Each site receives one portion of the mini-NVRF from every other site (covering the same portion of VIN's). Then each site merges these n files, detects duplicates, and prepares n files of duplicates to be removed from the NVRF. Each such file is then sent to the appropriate site. This process requires the handling of n squared tapes, but if n is small (say 10 federal regions) this should not be a problem.

7.5. Online Duplication Detection

Suppose that all duplicates have been successfully eliminated from the NVRF, and there is an update to the database. If we know the location of the last registration, we can readily remove any duplicate record that might arise. If we don't know the location of the previous registration (e.g. for new cars), then either we:

(1) Broadcast a query to all sites (expensive if there are many sites).

(2) Maintain a micro-NVRF which records only the VIN and the site of registration. Either the micro-NVRF is stored at one site[29] or it could be split it as described for the the mini-NVRF above. In either case we need only query the site where the micro-NVRF is stored, plus the site

where the vehicle is registered (if it was).

[29]It is fairly small (approx 3 gigabytes) and has a low access rate (mostly new car transactions, a fraction of all RCIP transactions).

8. Ration Check Address File Module

8.1. Generic Description

This module uses the most current NVRF to produce the address files for the Ration Check Production module. The NVRF records are used to create files with only the informaton required for ration check printing and mailing and reconcliation. These records are sorted as required for optimal handling by the USPS. The task will probably be handled over a period of 20 days[30] in order to spread the work load of the USPS, the CIPs, and the RCIPs over a period of time. It may be necessary to generate a unique identification code for ration check reconciliation in this module.

-- Inputs

The most current NVRF
The allocation algorithm

-- Outputs

Address files to be sent to the Ration Check Production module and to the Reconciliation module

-- Function

The primary function of this module is to create address files used for the printing and mailing of ration checks. Facilities for sorting the address files into proper sequence and providing staged address file subsets of a short period (20 days) are required. (See footnote below.) The address files are also used for ration check reconciliation, and may require the generation of a unique identification code for that purpose. This module also calculates the number of gallons allocated to each individual and prints it on each ration check, using a prescribed formula based on variables like vehicle category, weight, or state.

8.2. Proposal

We will dump the data base to tape every night. While we are doing this we will extract the appropriate portion [31] of the files to generate the next portion of the address file for the next mailing.

8.2.1. Integrated records

If the reconciliation record, and the continuation record (if any) are stored in the same sector as the vehicle registration record it is possible to immediately determine whether to issue a new check. The new check number can be generated and the sector rewritten on the next revolution of disk (while we are dumping the data base).

---

[30] Twenty days for a startup scenario, 60 for normal scenarios.

[31] Five percent of the database for the startup scenario, 1.7% for normal scenarios.

8.2.2. Separate Records

If the reconciliation record, continuation records and vehicle registration record are not stored in the same sector, it will not be possible to determine immediately whether to issue a new check. It will then be necessary to collect all portions of the relevant records for processing the following day.

The next day these records are sorted by VIN, merged, and new check numbers are generated for eligible vehicles. The extracted records will be updated while the database is being dumped to tape.

8.2.3. Sorting by ZIP

During slack periods the following day, the extracted files will be sorted by ZIP code and address.

9. Ration Check Production Module

The Ration Check Production module prepares the actual printed check on preprinted, controlled stock. The ration check may be a single or multiple form postcard, a self-mailer, or a check to be stuffed into an envelope. Whichever form is used, this module is responsible for maintaining a stock, controlling that stock, printing the checks as required and delivering the checks to the Mailing module. The address files will already be sorted appropriately, so that mechanical sorting of the printed checks will not be necessary. It may be the case that pre-encoded check numbers (using MICR or OCR techniques) will have to be read and recorded on the address files by this module for reconciliation purposes, in which case that function would not be done in the Ration Check Address File module.

-- Inputs

Ration check address files
Preprinted paper stock for ration checks

-- Outputs

Batches of printed checks to be sent to the Mailing module

(Possibly) An updated address file with encoded check numbers to be sent to the Reconciliation module.

-- Function

This module produces the printed ration checks, sorted into proper order, ready for mailing, handles all the required accounting for the paper stock, and possibly provides the Reconciliation module with the accounting files.

9.1. Proposal

We envision that the ration check issuance tapes will be sent out to another agency (Treasury) or contractor for printing. Whether or not the data processing is distributed should not affect the decision as to whether the printing of the checks would be centralized or distributed. However, if production of the check issuance tape and check printing are not close together some additional delay (1 day or less) may occur).

9.2. Printing Requirements

We intend that check numbers be generated and printed on the checks. At least some portion of the check number will not be sequential. Cf. 5.3.1 (15) for a description of the check number.

9.2.1. OCR

It would be possible to print the check numbers and gallonage on the check in OCR format. If offline reconciliation of checks is necessary, OCR readers could be used to recover this information. This is probably the prefered printing method.

9.2.2. MICR

Magnetic ink encoded characters could be used for the check number and gallonage. This would permit the use of industry standard MICR document readers for offline reconciliation. We have constrained check numbers to be all numeric, so alphabetics are not a problem. MICR printing is apparently more difficult and unreliable than OCR printing - especially when printing nonserial numbers.

9.2.3. Other Printing Techniques

It is possible to enter checks via keyboard to be reconciled off-line. This option would only be attractive if the majority of the checks were reconciled online and it was decided that returned checks did not need to be recorded.

It is doubtful that this option would be desirable, in view of the modest savings over OCR printing and the possibility of large costs being incurred if substantial offline reconciliation proved necessary.

9.3. Costs

The costs of check printing should be nearly invariant with respect to the organization of the data processing. See the Oak Ridge or Polk reports.

9.4. Credit Cards

The Electronic Funds Transfer (EFT) proposal, and possibly the automatic cash dispenser proposal as well, will require the production, stuffing and mailing of some 150M credit cards.

The entire annual U.S. credit card production capacity is only about 150 million cards. Six months to year would be required to double capacity.

Electronic Transfer of Funds costs are as follows:

Costs of Distributing
Credit Cards

Item	$/1K	Total $
Card	52	7.8M
Embossing	32	4.8M
Magnetic Encoding	50	7.5M

Note: We have assumed 150 million cards will be produced.

10. Mailing Module

The Mailing module receives presorted checks from the Ration Check Production module, performs accounting functions, and prepares the appropriate bundles to be delivered to the USPS. Nondeliverable checks will be returned to this module for handling. Close coordination with USPS will be required to minimize the work load on the postal delivery system.

-- Inputs

Printed checks from Ration Check Production

-- Outputs

Sorted checks to USPS
Nondeliverable checks to Reconciliation module

-- Function

Handle the accounting and control functions associated with delivery of ration checks in properly sorted and packaged bundles to the USPS, including accounting for nondeliverable checks for reconcliation.

10.1. Proposal

We expect that the mailing will be performed by the same outside agency or contractor that does the check printing. In any case, the organization of the data processing should not greatly affect either the costs or design of this module. See the Polk or Oak Ridge reports.

10.2. Credit Cards

For credit cards we have the following cost estimates:

Costs of Mailing
Credit Cards

Item	$/1K	Total $
Attaching to mailer inserting in envelope metering + mailing	50	7.5M
envelope	9	1.35M
postage	130	19.5M

Notes: We have assumed 150M cards will be produced. We also assumed 13 cents postage on presorted first class mail.

11. Coupon Issuance Points

11.1. Generic Description

The Coupon Issuance Points (CIPs) are responsible for exchanging valid ration checks for negotiable ration coupons. Because the coupons can be traded for money on the white market, the CIPs must be capable of the usual money-handling procedures, as is done in banks, post offices, etc. It is assumed that the CIPs will require identification consisting of a valid vehicle registration, a driver's license, and perhaps some other identification as well. Cashed ration checks are sent to the Reconciliation Module daily or weekly. Procedures may require the stamping of the vehicle registration to prevent fraudulent use in the RCIP module. The various tasks possible for CIPs include a wide range of activities, but it is felt that the CIPs should have simple procedures and should be easily accessible to the public. On the order of 40,000 CIPs are being considered.

-- Inputs

Ration checks, vehicle registrations, and other identification from entitled individuals
Controlled stocks of ration coupons

-- Outputs

Ration coupons to individuals
Ration checks returned to reconciliation module

-- Function

Exchange valid ration checks for ration coupons, performing the usual money-handling accounting and control procedures.

11.2. Scenario

11.2.1. Offline

The customer presents both a ration check and his/her vehicle registration and driver's license to a CIP teller. The ration check contains an encoded description of the vehicle for which it is intended. If the three documents are consistent, the teller records sufficient identifying owner information on the check and dispenses the ration coupons. Both the teller and the customer sign the ration check.

11.2.2. Online

The customer presents both his/her car registration, driver's license and ration check to the CIP teller. The teller enters the check number, the numeric portion of the VIN (or the license plate)[32] and the allotment. If all

[32] If alphanumeric terminals are available at the CIPs, the entire license plate number would be entered.

are contained on the same NVRF record, the teller gets a display of the two digit random response number, which should be checked against identical figures on the ration check. The teller and the customer sign the check, and the teller dispenses the coupons. The CIP identification, the teller identification, the current date, and, if used, the current time, are entered into the NVRF record.

If the check is from another region, the teller must preface the check number with a region number. The check number will then map to the correct regional data base.

There are several circumstances under which the CIP teller will direct the customer to an RCIP office:

(1) The special procedures byte is set, indicating some unresolved irregularity in the previous quarter's check cashing transaction. The teller will be informed, and the customer must go to an RCIP for manual reconciliation, e.g., a signature comparision.

(2) The CIP identification, teller identification and a date have already been entered for this quarter, i.e., someone has already cashed a check with the same check number-VIN combination.

(3) The check number, license number and VIN do not appear on the same NVRF record.

(4) Used, new, and unrecorded vehicles will require RCIP attention.

11.3. Number of CIP teller windows.

At three minutes per CIP transaction, twenty transactions can be completed during one hour. Therefore the number of CIP teller windows needed to carry the peak transaction load for all four scenarios[33] is as follows:

(1) Worst case startup schedule:
180K teller windows

(2) Best case startup scenario:
47K teller windows

(3) Worst case normal schedule:
60K teller windows

(4) Best case normal schedule:
18K teller windows

11.4. CIP Reconciliation

11.4.1. Preventing a teller from disbursing more coupons than are indicated on the ration checks cashed in

Each teller is given a certain sequence of coupons at the start of his/her terminal session. At the end of the session, the sum of the value of the coupons remaining and the accumulated check allotments must match the

---

[33] Cf. 3.2 for the scenario assumptions. Note that we have assumed uniform population distribution across the country.

amount initially given out.

11.4.2. Insuring that coupons are not disbursed without entering the transaction into the computer

Each day the system will create a file containing the total number of coupons disbursed by a CIP teller window. A CIP manager may query the system to retrieve the total daily coupon disbursal for the CIP. If that total does not match a manual count of the allotment on all checks cashed, a log of the daily transactions may be obtained to determine which checks have not been entered. Assuming 18K to 180K CIP teller windows, the accumulators will consume between 36KB and 360KB of memory.

Two basic schemes may be used to implement the log check, one being more labor intensive for the CIP personnel and the other requiring storage and more CPU processing time during working hours to maintain.

For the former, the CIP manager simply queries each NVRF record by check number and receives a display of the check number, the VIN, the allotment, the date cashed and the teller identification. The manager will then know which checks have not been entered, and, since checks and leftover coupons are returned together by the teller, which teller is responsible for the omission. It is estimated that there will be between 38 and 125 daily transactions per CIP teller window[34], i.e., that number of checks for the manager to enter. The process requires one disk seek for the check number lookup, but that is done after working hours (assuming single shift operations).

Under the latter scheme, rather than entering each check number, the manager would ask for a serial display of the check cashing information[35] and match it against the checks in hand. For this method it would be convenient to have the transactions either accumulated by teller or sorted by teller before the display, since otherwise they would appear in the order of their occurrence. The latter scheme involves, for each CIP teller window, a system of pointers linking the daily transactions. Its implementation would require a series of ten byte entries,[36] one for each transaction, a four byte pointer to the location for the next transaction entry, and a four byte pointer for each CIP teller window indicating the location of its last entry. Assuming that there are 2.25M to 6.75M transactions per day, depending on a sixty or

---

[34] The variation in the number of daily CIP teller window transactions is due to the difference in the number of CIP teller windows required for peak transaction loads under the four schedule scenarios. Cf. 3.2. for the scenario assumptions and 11.3 for the CIP figures. The number of daily transactions for all four scenarios are:

38 for worst startup,
24 for best startup,
38 for worst normal,
125 for best normal.

[35] The check number, the date of the transaction, the CIP/RCIP and teller identifications, and the quarterly gasoline allotment. Cf. 5.3.1 (15) through (19) for a definition of the data elements.

[36] Six bytes for the disk, sector and record numbers of the NVRF record in a transaction and four for the pointer to the previous entry for a particular CIP teller window.

twenty day processing schedule. The daily transaction log will be between 22.5MB and 67.5MB. The pointers to each last CIP window entry will take up between 180KB and 720KB, assuming 18K to 180K CIP teller windows.[37] Periodically the daily transaction logs are dumped to tape.

11.5. Terminal Configurations

Below we discuss several plausible terminal configurations for each CIP teller window.

11.5.1. Credit Verification Type Terminal Configuration

We envision a terminal similar to those used for credit verfication. It would have a numeric keyboard and 15-20 digit numeric display (probably light emitting diodes). It would also have several special function keys. Thus the terminal would look like a calculator.

The terminal would also contain a built-in 300 baud modem and a Data Access Arrangement (DAA) to connect to the phone line. It is expected that the terminals will be operated on polled multidrop private lines.

Terminals of this type could be purchased for $400 each. Maintenance costs have been estimated at 1% of acquisition cost per month.

Cost of CIP credit verification
type terminals required for peak load

Scenario	number	$ cost	Maintenance $ cost/month
Worst Case Startup	180K	72.0M	720K
Best Case Startup	47K	18.8M	188K
Worst Case Normal	60K	24.0M	240K
Best Case Normal	17.85K	7.14M	72K

11.5.2. Transaction Telephone Terminals

Bell Telephone leases a telephone with[38]

(1) a touch tone numeric pad

(2) a 8 digit display

(3) a magnetic strip credit card reader

(4) a touch tone decoder

These devices could be used as a CIP terminal on a public (dialup) telephone network. The CIP teller would insert a credit card which would dial up the computer. He would then insert a credit card from the client containing the check number and amount and enter the last few digits of the VIN.

The advantages of the transaction telephone are:

---

[37] Cf. 11.2 for the counts for each scenario.

[38] Other vendors manufacture similar terminals.

(1) possible reduction in communications cost

(2) faster, hence reduced labor costs and possibly some reduction in the number of terminals needed.

(3) fewer errors

(4) somewhat more secure against naive forgers

These terminals are presently tariffed at approximately $30 per month. Each requires a separate phone line.

Cost of CIP transaction telephone
type terminals required for peak load
(including maintenance)

Scenario	number	cost $/mo
Worst Case Startup	180K	5.4M
Best Case Startup	47K	1.4M
Worst Case Normal	60K	1.8M
Best Case Normal	17.85K	.54M

11.5.3. Automatic Teller Machines

These devices (made by IBM and others) accept magnetic strip credit cards, plus numeric key input, and dispense two denominations of bills (in our case ration coupons).

The devices presently cost $15,000 each. They would eliminate many CIP tellers and could remain open 16 hours (or more) per day. Thus fewer terminals would be needed.

Normal practice is to mail out credit cards and passwords in separate mailings. Customers are required to present the credit card to the credit card reader and to key in the password (sometimes known as PIN - Personal Identification Number). This system is in use successfully by commercial banks.

The lack of a signature offers opportunities for fraud. Vehicle owners could deny that they received the credit card and password, suggesting that someone else had obtained the credit card and password and used it to take the coupons. Without a signature it would be difficult to prove non-receipt. A similar problem arises after the system is in operation, in that a vehicle owner could claim that the credit card and password were stolen. But in this case, the public could be warned that they should keep the password and credit card apart and that losses resulting from simultaneous theft would not be made good.

The alleged mail intercept fraud would be attacked by spacing the two mailings far enough apart (one month) and demanding that the public report any claim of nonreceipt of credit card before the passwords are mailed out.

It is not clear that the public would accept the reporting demands, or the denial of claims due to simultaneous theft of credit card and password. Gas rationing, unlike bank credit cards, is not a voluntary system. Furthermore the risk of fraud will be greater, since bank credit cards are normally only sent to citizens with excellent credit ratings. Such persons have less incentive and more to lose by engaging in petty fraud than other segments of the population.

One possible solution is to build a hybrid system, in which vehicle owners could choose whether they preferred checks or credit cards with the attendant regulations.

Clearly automatic teller machines will be cheaper than human tellers if gas rationing continues for a long period of time. However, to construct and install this number of machines would require that large sums of capital be committed several years before the system went into full operation. Such a decision would only be warranted if DOE were fairly certain that gas rationing would be adopted.

Cost of CIP automatic cash dispensing terminals required for peak load

Scenario	number	$ cost	Maintenance $ cost/month
Best Case Startup	47K	705M	.71M
Best Case Normal	17.85K	268M	.27M

11.5.4. Labor Costs for CIPs

We have 135M CIP transactions which will require 3 minutes each to process. Clerical labor and overhead is estimated at $10.50 an hour[39]. Thus we will require at least 6.75M man hours ($71M) of labor per quarter.

If we staff teller windows to accommodate peak loads at any time we would double these figures for the best case scenarios, and quadruple them for the worst case scenarios. Variable staffing (by time of day) would probably be desirable.

[39] This figure is taken from the Oak Ridge Report, Section 1.4.1.

12. RCIP Scenarios and Terminal Configurations

12.1. The General RCIP Scenario

RCIP activity consists of all ration check redemption operations other than the normal situation of showing all proper identification and receiving coupons for a vehicle for which no coupons have been previously dispensed for the current quarter.[40] The identification requirements are more flexible for RCIP than for CIP transactions. The vehicle identification or a temporary facsimile issued to new cars as well as a driver's license are always mandatory; but a customer may not always be in possession of a ration check. In all RCIP cases a search for the NVRF record is accomplished through an index on either the VIN or the license plate number[41], the special procedures byte is set to a value indicating the irregularity which has occurred[42], and, just as in CIP transactions, the RCIP teller and the customer sign the new check. If there has never been an irregularity for a particular VIN and if the customer has a proper vehicle registration and driver's license, he/she is given the number of ration coupons indicated in the allotment data element in the NVRF record or calculated from the vehicle classification, regardless of the transaction difficulty.

The same RCIP activities take place, and the same data elements are collected, whether the RCIP processing occurs online or offline. If online, the entire transaction is completed while the customer is at the teller window. If offline, the information checking and recording aspects are performed via batch processing.

12.2. Specific RCIP Scenarios

12.2.1. The customer has not received a ration check

The new check number used, the current date, the RCIP identification and the teller identification, along with any information about the vehicle or customer which differs from the NVRF record, is recorded. If the NVRF record is contiguous, the data is written onto a continuation record; if the check cashing portion is physically separate from the owner and vehicle identifying portion, the information is split between a new record in the former and a continuation record in the latter.

12.2.2. New cars

A completely new NVRF record is created; all cashing information is recorded. Often the license plate number for the new vehicle is not available for up to six weeks after the date of purchase. If the RCIP index is based on this data element, the new owner will have to return to an RCIP to update the vehicle's record.

---

[40] For a description of the "normal situation", i.e., CIP ration check redemption, Cf. 11.1.

[41] Alternative indexes exist because of still unresolved disadvantages to both data elements. The reliability of the VIN is in question and license plate numbers are issued up to six weeks after the purchase of a vehicle.

[42] Cf. 5.3.1 (21) for a description of the data element.

12.2.3. Used cars

A new set of NVRF vehicle and owner information and check cashing information is recorded on a continuation record.[43] The former owner's data will be eliminated after reconciliation with the DMV. There will be no transferability of the ration check and no pro rata dispensing of ration coupons for used cars. It will be assumed that coupon negotiation between seller and buyer will have been part of the sale. The only exception will be if the seller has not cashed the ration check for the quarter. In this case the check will be voided, and a new check issued to the buyer.

12.2.4. Check number/VIN mismatch which was detected at a CIP

The differing check number and any information about the vehicle or owner which differs from the original NVRF record is recorded on a continuation record.

12.2.5. No vehicle record in the NVRF for the VIN

The procedure followed is identical to that for new cars. (12.2.2).

12.2.6. The quarterly check has already been cashed for the VIN

This is the only scenario in which no ration coupons will be dispensed on the first occurrence.

12.2.7. Any of the above situations occurring more than once per quarter

Presumably the DMV has reconciled any conflicting or duplicate data. The owner's identification must match the information in the NVRF record; if it does not, no coupons will be dispensed. The vehicle owner must rectify the DMV's vehicle records.

12.3. Number of RCIP Teller Windows Needed

Assuming ten minutes for each RCIP transaction, six transactions can be completed in one hour. At this rate the number of RCIP teller windows needed to handle peak transaction loads for all four schedule scenarios[44] is as follows:

(1) Worst case startup scenario:
66.7K teller windows

(2) Best case startup schedule:
17.3K teller windows

(3) Worst case normal schedule:
22K teller windows.

(4) Best case normal schedule:
6.67K teller windows.

---

[43] or split between a continuation record and a new transaction record. Cf. 5.4.5.

[44] Cf. 3.2 for a scenario description.

12.4. RCIP Reconciliation

Teller fraud would be controlled using the system which was described for CIPs in section 11.4. The latter of the two schemes in 11.4.2 would add from 13KB to 133KB[45] for accumulator storage, 2.5MB to 7.5MB[46] for the transaction log, and from 27KB to 267KB[47] for RCIP pointers.

12.5. RCIP Terminal Configuration

We envision that each RCIP teller will have an intelligent CRT terminal, with screen editing, and builtin encryption box and 1200 baud full duplex modem to query and update the NVRF.

Encryption is needed to satisfy security and privacy issues and intelligence will be required to manage the encryption box. The use of a buffered terminal with screen editing on a multi-drop line should provide savings on communications costs. In addition, local editing will protect the clerk from possible slow response from the computer.

A terminal of this type would cost about $3,000 apiece. None are presently on the market.

Cost of RCIP CRT terminals
required for peak load

Scenario	number	purchase cost in $	maintenance cost in $/mo
Worst Case Startup	66.7K	200.0M	1.33M
Best Case Startup	17.3K	51.9M	.43M
Worst Case Normal	22.0K	66.0M	.44M
Best Case Normal	6.67K	20.0M	.17M

Assumptions: $20/month maintenance charge per terminal during normal working hours (worst case schedules). $25/mo for extended hours (best case schedules). $20/month is what DEC charges to maintain a VT100 terminal.

12.6. Labor Costs for RCIPs

These costs are the similar to those for CIPs except that there are approximately 15 million transactions, 10 minutes each, yielding 2.5 million man hours ($26 million) per quarter.

---

45 depending on the number of RCIP teller windows. Cf. 12.3.

46 depending on the daily transaction load, which in itself is a factor of the mailing schedule.

47 again, depending on the number of RCIP teller windows. Cf. 12.3.

13. Reconciliation Module

The Reconciliation module provides for detailed ration check reconciliation, duplicate payment detection, fraudulent check alteration and replicaton detection, and update information for the Data Integration/NVRF module. A report (or file) is sent to the Audit/Enforcement module for handling exceptions. The reconciliation procedure involves matching cashed checks with mailed checks. In cases where mailed checks are cashed and RCIP registration exists for the same vehicle, Audit/Enforcement is notified of a possible fraud. In cases where mailed checks are not cashed after a certain period, that record may be removed from the NVRF under the assumption that the vehicle no longer exists, or at least no longer requires ration rights. The reconciliation process is the only mechanism for detecting and reducing fraud.

-- Inputs

Ration Check Address File
Nondeliverable checks
Cancelled checks
RCIP registration forms

-- Outputs

Update/correction files to Data Integration/NVRF
Exception cases to Audit/Enforcement
(Possibly) Usage data to Allocation

-- Function

Reconcile cashed checks with mailed checks and provide fraud detection and NVRF updates and corrections.

The Audit/Enforcement and Allocation modules are outside the scope of this study. Only the interfaces to the Reconciliation module need concern us here, and the fact that the Allocation module provides input to the Ration Check Address File module for determining the number of gallons allocated to each individual according to the allocation algorithm.

13.1. Proposal

13.1.1. Online CIPs We expect to perform reconciliation online at CIPs. See the chapter on Coupon Issuance Points and the chapter on File Organization.

Clearly some checks will be processed offline. This could either be keyed (as if from a CIP) or read by document readers (if MICR or OCR printing was used). The data processing would proceed as if the check were being cashed at a CIP.

13.1.2. Offline Reconciliation

Offline reconciliation (if CIPs are not online) is quite similar to that for centralized batch systems. See the Polk and Oak Ridge reports.

The reader should note that we do not need to physically sort the checks. It will be sufficient to have an index to the location of the check (or its photograph) should fraud investigations require it be examined.

At most one might run checks twice through a document reader. Once to record the check numbers (and amounts). This data could be sorted and checked against the check issuance tape. On the second pass through the document reader, bad checks could be extracted.

14. Networking

14.1. Network Configuration

14.1.1. Intersite Configuration

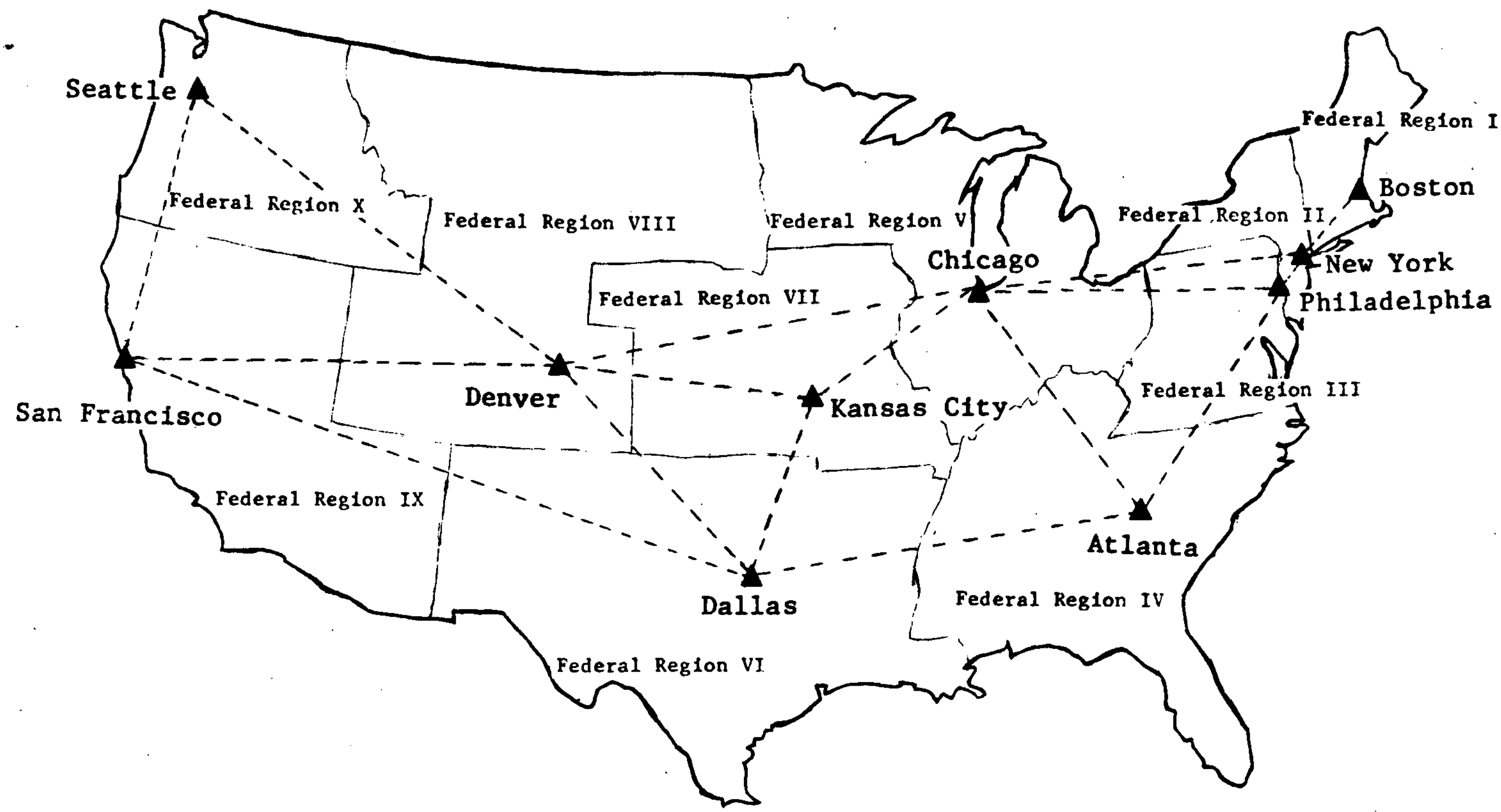
The network is configured as ten computer sites, one in each federal region. These sites are inter-connected with 17 links, one link between each pair of adjacent regions, with an additional link between New York and Chicago (to prevent all New England inter-site traffic from having to go through Philadelphia to the rest of the country). This configuration is shown on the map on the next page; Airline Mileages (the basis for line charges) are shown below:

Boston - N.Y.	220
N.Y. - Philadelphia	90
N.Y. - Chicago	750
Philadelphia - Atlanta	670
Philadelphia - Chicago	670
Atlanta - Chicago	590
Atlanta - Dallas	730
Atlanta - Kansas City	690
Chicago - Kansas City	420
Chicago - Denver	920
Dallas - Kansas City	450
Dallas - Denver	560
Dallas - San Francisco	1510
Kansas City - Denver	560
Denver - San Francisco	980
Denver - Seattle	1050
San Francisco - Seattle	750
17 lines	11,610

Figure 14.1

14.1.2. Regional Configurations

From each site, a number of lines will go to concentrators which are located throughout each region, as illustrated schematically in Figure 1. From the concentrators, multi-drop lines will be run to the individual terminals.


Seattle
Federal Region X
Federal Region VIII
Federal Region V
Federal Region I
Boston
Federal Region II
New York
Chicago
Philadelphia
Federal Region VII
Denver
Kansas City
San Francisco
Federal Region III
Federal Region IX
Atlanta
Dallas
Federal Region IV
Federal Region VI

Map 1

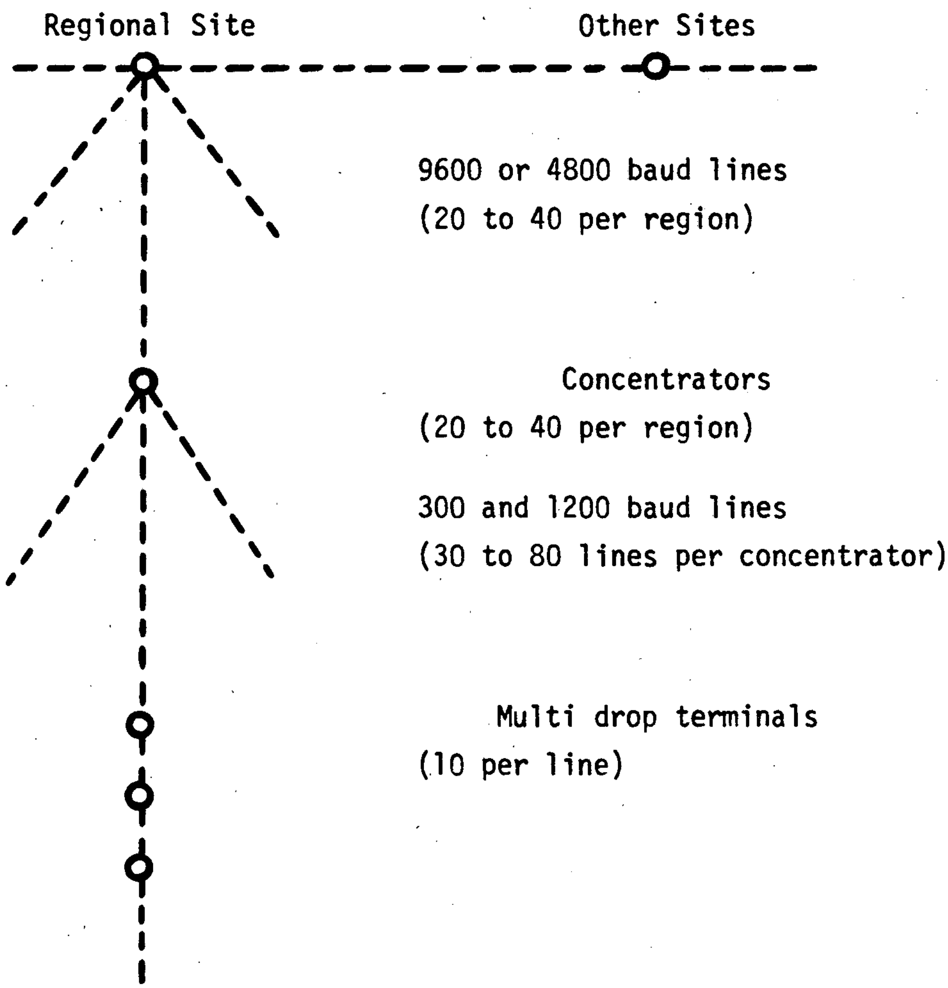

Regional Site
Other Sites
9600 or 4800 baud lines
(20 to 40 per region)
Concentrators
(20 to 40 per region)
300 and 1200 baud lines
(30 to 80 lines per concentrator)
Multi drop terminals
(10 per line)

Figure 1.

The number of concentrators is chosen to minimize line costs as described in Section 14.6. The cost of concentrators is assumed to be approximately $600 per line for CIP terminals and $1000 per line for RCIP terminals. (Typical concentrator costs are $400 to $600 per line. The upper end of this range is used since multi-drop lines are somewhat more complex to interface. For RCIPs the cost includes local intelligence for automatic forms generation for the terminals.)

Multi-drop lines are assumed for the terminals since this will provide significant savings in mileage and termination charges. For up to ten terminals, CIPs will receive acceptable service from 300 baud lines and RCIPs will receive acceptable service from 1200 baud lines. The marginal cost reduction achieved in multi-drop lines is largest for the first few terminals per line and is very small beyond 10 terminals per line. We will consider 10 terminals per line in this study.

14.2. Transaction Sizes

14.2.1. CIP Transactions

Each CIP transaction requires one message and one reply. CIP messages are assumed to contain 20 characters of information from terminal to site and 5 characters from site to terminal. CIP messages from site to site and from site to concentrator are assumed to require 18 characters of overhead for each message in each direction. This includes 9 bytes of header and 9 bytes for acknowledgement for each message. This is a minimum value estimated from the X.25 protocol. Each CIP transaction will thus generate 38 characters of traffic inbound. CIP messages between concentrator and terminal are assumed to require 30 characters of overhead per message. For 3 minute transactions, ten terminals per line, each line has 3.3 transactions per minute, requiring 165 characters (in each direction) over a line capable of 30 * 60 = 1800 characters per minute. This is less than 10% line utilization and will provide good response. Overhead between concentrator and terminal is not critical since these lines are not near saturation.

14.2.2. RCIP Transactions

Each RCIP transaction is assumed to consist of an inquiry (small message), a reply (large message), an update (large message) and a response (small message). Small messages are assumed to be 50 characters and large messages are assumed to be 400 characters. Including overhead (18 characters/message), each RCIP transaction will generate 486 characters of traffic in each direction. This figure applies for site to site and site to concentrator messages.

For concentrator to terminal transactions we must add in the effects of screen formatting which is supplied by the concentrator. We assume that this will double the actual number of characters sent to the terminal and back, i.e. 1000 characters in each direction. If 10 RCIP terminals share a multi-drop line and each generates a transaction every 10 minutes then the transaction rate per line is 1 per minute or 1000 characters/minute. The line capacity is 120 characters/second * 60 = 7,200 characters/min. This is 14% line utilization and will provide good response.

14.3. Intersite Traffic

We arbitrarily assume that 10% of all transactions will involve intersite communication. (The total communications cost is not very sensitive to intersite traffic rates, since the intersite network is only a small part of the total network.) We assume each transaction requires only one hop, since all adjacent sites are directly connected. Multi-hops are allowed but are assumed to be only a very small fraction of the single hop traffic and are ignored here.

14.4. Communication Costs

Since the terminals in the network are widely dispersed, few links could economically use lines of capacity higher than 9600 baud. Our cost estimates are based on 9600 baud (or less) lines exclusively. These lines cost $.52/month per mile and $40/month per termination. 9600 baud modems cost $256/month per termination. 4800 baud modems cost $135/month per termination. 2400 baud modems cost $60/month per termination. 1200 baud modems cost $42/month per termination. 300 baud modems cost $12/month per termination.

14.5. Intersite Communications

Intersite mileages are shown in Table 14.1. Seventeen links are proposed.

Character rates are derived from section 3.2.5 with CIP transactions requiring 38 characters and RCIP transactions requiring 486 characters. For example, worst case startup (WCS) CIP transactions requires

1000 (transactions/sec.) * 10% * 38 = 3800 character/second.

If these transactions are evenly distributed over 17 links then each link requires

3800/17 = 224 characters/second

for the worst case startup. Each link can support 9600/8 = 1200 characters/second. We will assume a 50% line efficiency and use 600 characters/second as a line capacity. Then the worst case startup CIP transactions require 224/600 = 37% of line capacity.

Intersite Communications Traffic at Peak Load (17 links)

	WCS	BCS	WCN	BCN
CIP Characters/Second/line	224	58	75	22
RCIP Characters/Second/line	314	83	106	31
Total Characters/Second/line	538	141	181	53
Baud rate required	9600	2400	4800	1200
% of line capacity	89%	96%	60%	72%
Cost/year	$192K	$112K	$143K	$105K

The cost for mileage and terminations does not vary with line speed up to 9600 baud. Thus:

Mileage	11610 * .52 * 12	= $72K/year
Terminations	17 * 2 * 40 * 12	= 16K/year
		88K/year

To this figure must be added the cost of modems, e.g.

Modems (9600) 17 * 2 * 256 * 12 = $104K/year

If the regional computers were centralized, with each regional site becoming a super-concentrator, then the intersite communications costs would be as follows (assuming Kansas City as a central site):

Mileages (Airline)

Kansas - Boston	1260
Kansas - N.Y.	1110
Kansas - Philadelphia	1050
Kansas - Atlanta	690
Kansas - Chicago	420
Kansas - Dallas	450
Kansas - Denver	560
Kansas - San Francisco	1540
Kansas - Seattle	1520
9 links	8600

Inter-Regional Communications Costs for Centralized Computers (9 Links)
With Regional Concentrators

	WCS	BCS	WCN	BCN
CIP				
Char/Sec/Region	3800	988	1265	376
Lines/Region	6.3	1.6	2.1	.6
RCIP				
Char/Sec/Region	5346	1409	1798	535
Lines/Region	8.9	2.3	3.0	.9
Total				
Char/Sec/Region	9146	2397	3063	911
Lines/Region	15.2	4.0	5.1	1.5
Cost/Year	$1,856K	$464K	$696K	$232K

For each set of lines the cost is:

mileage	8600 * .52 * 12	=	$54K
termination	9 * 2 * 40 * 12	=	7K
modems	9 * 2 * 256 * 12	=	55K
TOTAL			$115K

Multiply by the number of lines required to obtain the total cost. We would expect some economies of scale but cannot quantify them here. Non-recurring costs (regional concentrators) are not included in this estimate, but would be approximately 10% of the annual recurring costs.

For some configurations, the concentrator lines could be run directly to a central location without using regional concentrators. In that case, no additional terminations, modems, or concentrators would be required, only the mileage charges would apply. Mileage costs for this plan are $54K per line. The number of lines required is taken from the site to concentrator analysis.

Inter-Regional Communications Costs
for Centralized Computing
Without Regional Concentrators

	WCS	BCS	WCN	BCN
Char/Sec/Region	9146	2397	3063	911
Lines/Region	33	13	16	7
Added Cost/Year	$1,782K	$702K	$864K	$378K

14.6. Number of Concentrators Needed

For mathematical tractability, we assume a uniform population distribution, equal areas and population in each of the regions, compact shapes for each of the regions, and regional sites at the centroid of each region. (Alaska and Hawaii, especially, violate these assumptions.)

For each regional site, let N be the number of concentrators and T be the number of terminal multi-drop lines (=number of terminals /10), which is given in Section 3.5. For a region of area A, the average length of N lines from the center to cover the area uniformly at N locations is approximately 0.35 * $\sqrt{A}$. (The area of the United States is approximately 3,500,000 square miles.)

In order to select the number of concentrators, N, we seek to minimize those costs which depend on N.

The regional costs (which depend on N) are:

Site to Concentrator:

N * (mileage costs + termination costs)

Concentrator to Terminal:

T * (mileage costs)

For concentrator to terminal communications the termination and concentrator port costs do not vary with the number of concentrators. The mileage varies as a function of area covered, which is in turn a function of the number of concentrators N.

Site to Concentrator:

$N * (.35 * \sqrt{A} * .52 + 2 * 40 + 2 * 256)$

Concentrator to Terminal:

$T * (.35 * \sqrt{A/N} * .52)$

A = regional area = 350,000 square miles

$\sqrt{A}$ = 590 miles

total variable costs = $N * (108 + 592) + T * 108 / \sqrt{N}$

differentiating and setting to zero:

700 + T * 108 * (-1/2) N**(-3/2) = 0

N**(-3/2) = 700/(T*108*1/2) = 13/T

N = (T/13)**(2/3)

If 4800 baud lines are used then the optimum selection of N is

N = (T/8.48)**(2/3)

For T1 CIP lines and T2 RCIP lines with N concentrators, the required line speed is

$$\frac{\text{transactions/sec * characters/transaction * 8 bits/char.}}{\text{50\% line utilization}} \Big/ N$$

where transactions/sec =

$$\frac{\#\ \text{lines} * 10}{\#\ \text{seconds/transaction}}$$

Substituting, required line speed =

$$\left(\frac{\frac{T1 * 10}{180} * 38 * 8}{.5} + \frac{\frac{T2 * 10}{600} * 486 * 8}{.5} \right) \Big/ N$$

= (33.777 * T1 + 129.6 * T2) / N

14.7. Regional Site to Concentrator to Terminal Communications

The regional communication costs are composed of the following elements:

concentrator lines	
mileage charges	$ .52/mile/month
local loop charges	$40/month
modem charges	$256/month
concentrators	
concentrator ports	$600 each
CIP ports	$600 each
RCIP ports	$1000 each
terminal lines	
mileage charges	$ .52/mile/month
local loop charges	$40/month
CIP modem	$12/month (300 baud)
RCIP modem	$42/month (1200 baud)

Terminal line costs will not include modem costs at the terminal ends, as these costs are included in the terminal costs.

From Section 14.6, site to concentrator lines cost mileage plus local loop plus modem costs:

N (.35 * $\sqrt{A}$ * .52 + 2 * 40 + 2 * 256) =
N * $700/month = N * $8,400/year

or N * $458/month = N * $5500/year (4800 baud)

Where N is the number of concentrator lines per region.

Concentrators cost:

N * 600 + (# CIP lines) * 600 + (# RCIP lines) * 1000

The concentrator to terminal line cost consists of a concentrator to distribution point mileage for each group of 10 terminals plus local loop plus modem and a distribution mileage for each of 10 terminals plus local loops. CIP and RCIP terminal lines are treated separately.

For T1 CIP lines (=#CIP terminals /10) cost =

$$T1 * (108/\sqrt{N} + 40 + 12 + 10 * (108/\sqrt{T1} + 40))$$

for T2 RCIP lines cost =

$$T2 * (108/\sqrt{N} + 40 + 42 + 10 * (108/\sqrt{T2} + 40))$$

These are regional monthly costs.

Regional Communication Costs (9600 Baud lines)

Scenario	WCS	BCS	WCN	BCN
# Lines				
CIPs	1800	470	600	179
RCIPs	667	173	220	67
TOTAL	2467	643	820	246
# Concentrators N = (T/13)**(2/3)	33	13	16	7
Line speed required	4500	3000	3100	2100
Concentrator Line Costs (annual) N*$8,400/year	$277,200	$109,200	$134,000	$59,000
Concentrator Fixed Costs N*600+T1*600+ T2*1000	$1,800,000	$500,000	$600,000	$180,000
Terminal Line Costs (annual)	$15,000,000	$4,200,000	$5,300,000	$1,800,000
National Costs Per Year 10* (Concentrator line and terminal line costs)	$153,000,000	$43,000,000	$54,000,000	$18,000,000
National Fixed Costs 10* concentrator fixed costs	$18,000,000	$5,000,000	$6,000,000	$1,800,000

Regional Communications Costs (4800 baud lines)

Scenario	WCS	BCS	WCN	BCN
# Lines				
CIPS	1800	470	600	179
RCIPS	667	173	220	67
TOTAL	2467	643	820	246
# Concentrators N = (T/8.5)**(2/3)	44	18	21	9
Line speed Required	3400	2200	2400	1700
Concentrator Line Costs (annual) N*$5500	$242,000	$99,000	$116,000	$50,000
Concentrator Fixed Costs N * 600+ T1 * 600+ T2 * 1000	$1,770,000	$466,000	$593,000	$180,000
Terminal line Costs/year	$15,000,000	$4,200,000	$5,300,000	$1,700,000
National Costs Per year 10* (concentrator line + terminal line costs)	$152,000,000	$43,000,000	$53,800,000	$17,900,000
National Fixed Costs 10*Concentrator fixed costs	$18,000,000	$4,700,000	$5,900,000	$1,800,000

15. Hardware Configurations

In this chapter we will discuss the hardware configurations for the regional (or central) computer centers. We will discuss the matter in 5 separate sections.

(1) Workload estimation

(2) Central processors

(3) Disks

(4) Tapes

(5) Front End Processors for communications

15.1. Workload Estimation

We have estimated the CPU workload by comparing the proposed system to a typical airline reservations application running under IBM's transaction processing operating system ACP.

An average airline reservation transaction requires 10 disk seeks (including updating duplicate files) and some 15,000 instructions. IBM suggested that CPU requirements scale roughly linearly with the number of disk seeks. Most of the CPU instructions are systems code for communications and file system operation. Airline specifications require that 90% of all transactions be processed in under 1 second. There are IBM 3033's running in production environments at 215 transactions/second[48]. If the Hypervisor (a virtual machine monitor) is run, to permit batch operations under a copy of MVS on the same machine as ACP, then about 10% more CPU cycles are required by ACP for the same workload. We shall assume that we will run the Hypervisor on our production machines.

Running IBM's standard software (CICS/MVS) results in a ten-fold reduction in performance.

15.1.1. CIP and RCIP Transactions

An RCIP transaction should require about as much processing as an airline transaction. A CIP transaction is much simpler and it is estimated that it will require no more than .5 as much processing as an airline transaction.

We shall discuss only the case where the vehicle records are integrated with the check reconciliation records (physically contiguous). We shall also assume that vehicle records are stored according to a hash code calculated from an invariant portion of the check number. Thus CIP queries (on check number) can hash directly to the relevant record. RCIP queries on VINs must first hash to an index entry which points to the vehicle record[49]. An RCIP transaction will entail 11.2 seeks[50] as follows:

---

48 Without the Hypervisor. IBM estimates of capacity are 20% higher. We shall use the observed performance capabilities.

49 See the chapter "File Organization".

50 Assuming duplicate files, 9.2 seeks/transaction if no duplicate files are kept.

(1) 1.2 seeks to hash into the VIN index[51]

(2) 1.0 seek to retrieve the integrated vehicle registration and check reconciliation record.

(3) 3.0 seeks to reread and rewrite the vehicle record and its duplicate.

(4) 1.0 seeks to find the continuation of the vehicle record, assuming it is on a different page.

(5) 3.0 seeks to reread and rewrite the continuation of the vehicle record and its duplicate, again assuming a page different from the main record.

(6) no seeks to look up the city name as it will be kept in main memory.

(7) 1.2 seeks to look up the street name and number to get the zip code (See footnote)

(8) 0.8 seeks = reads generated by the 20% of all transactions which must be retried.

A CIP transaction will involve 3.2 seeks[52]:

(1) 1.2 seeks to find the check record (by hashing). (See footnote)

(2) 2.0 seeks to rewrite the check record (and its duplicate).

Note: The vehicle (and continuation) records must be reread because we can not afford to keep all of these disk sectors in memory for the ten minutes required for the teller to process the RCIP transaction. We have pessimistically assumed that we will always encounter a continuation record.

Disk Seeks per Transaction

Transaction Type	Seeks With Duplicate Files	Seeks Without Duplicate Files
RCIP	11.2	9.2
CIP	3.2	2.2

15.2. Central Processors

15.2.1. Type of machine

15.2.1.1. Minicomputers (16 bit)

At least one vendor (Tandem) sells a fault tolerant transaction processing system built of multiple 16 bit word size minicomputers. Several financial institutions use such systems (by Tandem and other vendors).

[51] We have allowed a generous .2 seeks for rehashing due to sector overflows.

[52] Assuming duplicate files are maintained, if not only 2.2 seeks/transaction.

For example, the Bank of America has a system with 9,000 terminals and 30-40 transaction/second capability.

One advantage of such a system is its modularity. These systems can be readily configured to handle different workloads in different regions. Incremental capacity increases are straightforward. Because of the multiplicity of processors the amount of hardware used as spares can be reduced.

There are several disadvantages to mini-based systems. In general the software supplied on such systems is not as good as that provided on large machines. Furthermore, the small address space, typically 65,000 bytes, will pose awkward problems for some applications programs. Programs written in IBM 370 assembler (such as VINA) will not run on these machines. Lastly, these machines generally do not have 6250 bpi tape drives which are needed to efficiently backup the data base and generate print tapes.

We expect to see transaction processing rates of 1-2.5 transactions/second per processor (assuming front end processors). This number comes from discussion with Bank of America personnel.[53] Assuming a $40,000/CPU cost based on typical TANDEM prices, this comes to about $20,000 per transaction/second.

15.2.1.2. Medium Size (32-bit) machines

Medium sized machines with 32 bit word sizes offer a large address space of several megabytes, thereby avoiding one of the most severe problems of the mini's. Machines in this class include the IBM 4300 series and the Digital Equipment Corporation (DEC) VAX-11/780 computer. There are a wide variety of IBM and IBM compatible machines in this class. Such machines would permit the use of programs such as VINA. However, IBM does not offer its main operating system OS/VS2 on its model 4300 machines, instead offering OS/VS1.

DEC claims that a VAX would handle 15 transactions/second. This comes to approximately $15,000 per transaction/second (assuming a VAX runs about $225,000). An IBM 4341 at $300,000 looks like it would handle 35 transactions/second[54] yielding a unit price of about $8,000 per (airline) transaction/second.

15.2.1.3. Large (32 bit) machines

IBM claims that a 3033 running ACP (Airline Control Program) will handle 200 airline reservations transactions /second[55]. A machine of this type would cost about $3.5 million. This comes to about $18,000 per transaction/second.

---

[53] The Bank of America transaction are predominently checking account transactions similar in complexity to a CIP transaction - although more information (i.e., the name of the account holder) is displayed to the teller. They run on General Automation 16 bit minicomputers.

[54] Scaled from 3033 performance, assuming Hypervisor is running.

[55] No such system has been run in production at over 215 transactions/second.

An Amdahl machine, the 470 V8, at about the same price appears to be able to handle 50% more traffic, yielding a unit price of $12,000 per transaction/second.

15.2.2. Stock Operating Systems

Using IBM's standard system (CICS atop the MVS operating system) requires 8 to 10 times the CPU capacity.

15.2.3. Summary

All of these estimates are fairly rough and will depend on the detailed specifications and design of the system. It does appear that minicomputers offer no great cost advantage and potential software problems. Among the 32 bit machines the optimal number and size of machines is unclear and we would expect a variety of bids on a RFP. In any case the costs appear to be somewhere between $8,000-20,000 per transaction/second. We shall assume $20,000 per transaction/second, i.e., a large IBM 3033[56].

15.2.4. CIP Transaction CPU Requirements

With no spare processors the failure of a single processor would substantially degrade performance at peak periods, depending on the number of processors. However, such a design is awkward if the processors are physically distributed, because we would probably have only two processors per site. Hence outages would have a severe impact.

CIP CPU Costs
(no spare processors)
(duplicate files)

Scenario	Under ACP		Under CICS/MVS	
	Purchase Cost in $	Maintenance Cost in $/mo	Purchase Cost in $	Maintenance Cost in $/mo
Worst Case Startup	10.0M	35K	100M	350K
Best Case Startup	2.6M	9.1K	26M	92K
Worst Case Normal	3.3M	11.7K	33M	117K
Best Case Normal	1.0M	3.5K	10M	35K

Assumptions: CPU cost per transaction/second capacity = $10,000 under ACP, $100,000 under CICS/MVS. Maintenance cost $35/month per transaction/second capacity under ACP, $350/month under CICS/MVS.

With 50% spare processors the failure of a processor would usually be masked by a spare processor. Such a design is probably necessary if the

---

[56] We have used the $20K/transaction estimate from the IBM 3033 because these machines are readily available while no IBM 4341's have yet been shipped and there is already a several year backlog of orders.

processors are physically distributed.

CIP CPU Costs
(50% spare processors)
(duplicate files)

Scenario	Under ACP Purchase Cost in $	Under ACP Maintenance Cost in $/mo	Under CICS/MVS Purchase Cost in $	Under CICS/MVS Maintenance Cost in $/mo
Worst Case Startup	15.0M	53K	150M	530K
Best Case Startup	3.9M	13.7K	39M	137K
Worst Case Normal	5.0M	17.5K	50M	175K
Best Case Normal	1.5M	5.3K	15M	53K

Assumptions: non-redundant CPU cost per transaction/second capacity = $10,000 under ACP, $100,000 under CICS/MVS. Maintenance cost $35/month per transaction/second of nonredundant capacity under ACP, $350/month under CICS/MVS.

15.2.5. RCIP Transaction CPU Requirements

With no spare processors the failure of a processor would substantially degrade performance (at peak periods) depending on the number of processors. Such a design is not practical if the processors are physically distributed.

RCIP CPU Costs
(no spare processors)
(duplicate files)

Scenario	Under ACP Purchase Cost in $	Under ACP Maintenance Cost in $/mo	Under CICS/MVS Purchase Cost in $	Under CICS/MVS Maintenance Cost in $/mo
Worst Case Startup	2200K	7.8K	22.0M	78K
Best Case Startup	580K	2.0K	5.8M	20K
Worst Case Normal	740K	2.6K	7.4M	26K
Best Case Normal	220K	.8K	2.2M	7.8K

Assumptions: CPU cost per transaction/second capacity = $20,000 under ACP, $200,000 under CICS/MVS. Maintenance cost $70/month per transaction/second capacity under ACP, $700/mo per transaction under CICS/MVS.

With 50% spare processors the failure of a processor would usually be masked by a spare processor. Such a design is probably necessary if the processors are physically distributed.

RCIP CPU Costs
(50% spare processors)
(duplicate files)

Scenario	Under ACP Purchase Cost in $	Under ACP Maintenance Cost in $/mo	Under CICS/MVS Purchase Cost in $	Under CICS/MVS Maintenance Cost in $/mo
Worst Case Startup	3300K	11.7K	33.0M	117K
Best Case Startup	870K	3.0K	8.7M	30K
Worst Case Normal	1100K	3.9K	11M	39K
Best Case Normal	330K	1.2K	3.3M	12K

Assumptions: nonredundant CPU cost per transaction/second capacity = $20,000 under ACP, $200,000 under CICS/MVS. Maintenance cost $70/month per transaction/second of nonredundant capacity under ACP, $700/month per transaction/second under CICS/MVS.

15.2.6. CPU costs incurred by distributed systems

Historically there have been sizable economies of scale associated with large centralized processing facilities. This is no longer always the case. At the present time it would appear that the smaller IBM 4341 has a better price/performance ratio than the 3330 on which our calculations are based. This is because the 4341 is a newer machine incorporating improved technology. The situation may reverse when the next generation of big machines is delivered (early 1981?). In any case computing costs should only go down from our estimates. We have not included any separate calculations for computing costs for distributed systems, as present pricing suggests that they will be lower if different at all.

We are hesitant to use IBM's optimistic characterization of 4341 in the absence of significant documented experience with ACP on the machine.

Within a single series of machines there are still economies of scale sometimes amounting to 25% in cost effectiveness, i.e., a distributed system might incur 25% higher costs for CPUs.

15.3. Disk Requirements

15.3.1. Disk Drive Size Considerations

The calculations for storage system configuration below are based on IBM 3350 disk drives. The disk drives are sold two spindles to a cabinet. The two spindles share the same read/write electronics. Each spindle has a nominal capacity of 317.5MB. When formatted with 1KB sectors, the drives have a capacity of 270MB per spindle. The average access time of drives is 33 msec (including latency). The peak transfer rate is 1.2MB/second. The drive uses the Winchester technology, that is it employs sealed non-removable packs. Such drives are considerably more reliable than earlier drives equipped with removable packs such as the IBM 3330. For reliability reasons we have excluded non-Winchester drives (commonly referred to as Storage Module Drives) from consideration.

Even if the level of storage provided by a 3350 disk drive were not required (e.g. because one needed many disk arms to perform many seeks/second) there is little point in buying smaller drives. The economics of manufacturing and marketing disk drives are such that smaller IBM compatible Winchester drives are not much cheaper. Smaller, less expensive, Winchester drives (not 370 compatible), are becoming available, but they generally have slower access times. Since the only reason for buying smaller drives is to gain more disk arms and hence more seeks/second capability, these new drives are not yet worthwhile.

15.3.2. Disk sector size

We shall assume that the disks are formatted with 1 kilobyte sectors, although this wastes some 15% of the disk surface. There are two reasons for the choice of sector size. One is that ACP only supports 381 byte and 1055 byte sector sizes. The other reason is that with rotational position sensing the sector size becomes the major determinant of disk controller and channel utilization. Since an IBM disk controller costs almost as much as 4 spindles and since the higher traffic scenarios are seek limited rather than storage limited, it is worthwhile to trade storage capacity for reduced channel and controller requirements.

15.3.3. Device utilization levels

In order to prevent excessive queueing delays for disk requests it is desirable to operate the disk drives at 50% utilization, i.e., 9 seeks/second. This is the design practice in the airline industry.

Heavy disk controller or channel utilization not only creates queueing delays, but also can degrade performance by requiring additional disk revolutions for transfers which were unable to complete because the controller or channel were busy when the sector to be transferred came under the recording heads. Buffering in the controller or drive can alleviate such problems. We shall design the storage system under the assumption that controllers and channels will be limited to 30% loading. Again this is common airline practice. Assuming 1K sectors, 1MB/second transfer rates and 9 seeks/second, 4 drives (8 spindles), which is the maximum allowed, can be put on a controller. Approximately 30 spindles can be put on each channel, i.e., 4 controllers with 8 spindles each.

15.3.4. Storage requirements

Storage Requirements

File	Single Copy Space	Single Copy 3350 Spindles	Two copies Space	Two copies 3350 Spindles
NVRF	27.5GB	102	55.0GB	204
VIN index	3.3GB	13	6.6GB	26
Street index	.2GB	1	.4GB	2
Total	31GB	115	62GB	230

Assumptions: 5 million streets; 35 bytes per street record which includes street name, upper and lower bound on house numbers, and zip code. See chapter on File Organization for further details. Disks are assumed to be formatted in 1KB sectors. Disk columns do not sum to total due to rounding errors.

15.3.5. Disk (unit) costs

We shall configure the storage system from IBM 3350's, hanging 4 drives (8 spindles) per controller. There are 3 types of drives relevant to us. Each controller must have at least one A2 type drive (it has some controller electronics). Incremental drives can be type B2 (no controller electronics). We also include one C2 type drive per controller (it can take over the functions of the A2 drive if the A2 drive fails).

Thus each controller will have one A2 drive, one C2 drive, and two B2 drives. Four drives (8 spindles) is the maximum number of drives allowed per controller.

Disk System
Component Costs

IBM 3350-A2 disk drive	$40K	$300/mo
IBM 3350-B2 disk drive	$32K	$225/mo
IBM 3350-C2 disk drive	$42K	$315/mo
IBM 3830-2 disk controller	$66K	$205/mo
controller + one A2 drive + two B2 drives + one C2 drive	$212K	$1270/mo
average per spindle	$26.5K	$160/mo

Note: controller costs include 2 channel switch; each drive contains 2 spindles; each spindle has 270MB capacity when formatted with 1KB sectors.

15.3.6. Numbers of spindles required

Disk Spindles Required
(non-duplicated files)

Scenario	Spindles Needed				
	CIP	RCIP	Both	Storage	Net
Worst Case Startup	244	113	357	115	357
Best Case Startup	64	30	94	115	115
Worst Case Normal	81	38	119	115	119
Best Case Normal	24	11	35	115	115

Notes: CIP column gives number of spindles required to meet disk seek traffic generated by CIP transactions. Similarly for RCIP column. "Both" column is sum of first two. The "Storage" column gives the number of spindles required to store one copy of the files. The "Net" column is the minimum number of spindles which meet both the traffic requirements and the storage requirements.

Disk Spindles Required
(duplicated files)

Scenario	Spindles Needed				
	CIP	RCIP	Both	Storage	Net
Worst Case Startup	356	138	494	230	494
Best Case Startup	92	36	128	230	230
Worst Case Normal	118	46	164	230	230
Best Case Normal	35	14	49	230	230

Notes: CIP column gives number of spindles required to meet disk seek traffic generated by CIP transactions. Similarly for RCIP column. "Both" column is sum of first two. The "Storage" column gives the number of spindles required to store both copies of the files. The "Net" column is the minimum number of spindles which meet both the traffic requirements and the storage requirements.

15.3.7. Aggregate Disk Costs

Disk System Costs
(nonduplicated files)

Scenario	RCIP only purchase $ cost	RCIP only maintenance $cost/mo	RCIP and CIP both purchase $cost	RCIP and CIP both maintenance $cost/mo
Worst Case Startup	3.05M	18.4K	9.46M	57.1K
Best Case Startup	3.05M	18.4K	3.05M	18.4K
Worst Case Normal	3.05M	18.4K	3.15M	19.0K
Best Case Normal	3.05M	18.4K	3.05M	18.4K

Note: Only the worst case scenarios with CIPs online are ever traffic limited. All other cases are storage limited.

Disk System Costs
(duplicated files)

Scenario	RCIP only purchase $ cost	RCIP only maintenance $cost/mo	RCIP and CIP both purchase $cost	RCIP and CIP both maintenance $cost/mo
Worst Case Startup	6.1M	36.8K	13.1M	79.0K
Best Case Startup	6.1M	36.8K	6.1M	36.8K
Worst Case Normal	6.1M	36.8K	6.1M	36.8K
Best Case Normal	6.1M	36.8K	6.1M	36.8K

Note: Only the worst case startup scenario with CIPs online is ever traffic limited. All other cases are storage limited.

In the tables above we have excluded costs for spare drives and controllers, and for drives and controllers used for batch operations. Since ACP and MVS can not share channels, etc., it would be necessary to purchase separate controllers and drives for the batch (mailing) work. The number of drives and controllers needed for batch and spares will depend on the machine configuration and the way in which the workload and equipment is shared between machines. Every machine should have at least one spare controller and drive accessible to it; and another controller and 2 drives for batch work is probably desirable. Possibly the batch drives could serve as the spares for the online system.

Shown below are the costs for batch and spare disks for a physically distributed system. For a physically centralized system, the costs would be somewhat less and depend on the number of processors used.

Total Cost of Spare and Batch Disks
(10 regional sites)

Items	Purchase $cost	Maintenance $cost/mo
Batch Disks one controller +one A2 drive +one C2 drive (per site)	1.48M	8.20K
Spare Disks one controller +one A2 drive (per site)	1.06M	5.05K

Notes: controller costs include 2 channel switch; each drive contains 2 spindles; each spindle has 270MB capacity when formatted with 1KB sectors. Costs shown are totals for all 10 sites.

15.3.8. Alternative Mass Storage Devices

Alternatives such as an automatic tape library, or a device such as an IBM 3850[57] or CDC 38500[58] were rejected because they could not support the number of random accesses generated by this application.

Optical disks were rejected because they are not yet commercially available, and they would not support the access rates for the bigger scenarios.

Bubble memories were rejected due to lack of commercial availability, as standard peripherals, and expense.

Charge coupled device memories and random access semiconductor memories are too expensive to use for the entire NVRF. Furthermore, it is doubtful that sufficient quantities would be available in the required time frame. If the reconciliation records were segregated from the vehicle registration records, then such devices would be more attractive for the reconciliation files. Nonetheless they are not presently economic (by a factor of ten).

Sometime in the next five years a storage system configured of magnetic bubbles for check reconciliation records and larger magnetic or optical disks for vehicle records may become viable. For the present we shall configure the system with moving head disk drives.

---

[57] This device is a jukebox of cartridges of magnetic tape which are recorded with a helical scan (like videotape).

[58] This device is similar to the IBM device, but smaller and cheaper. It does uses longitudinal recording (like a tape drive).

15.4. Tapes

The major demand for tape drives is for dumping the database to tape every night and writing out the address file. We want to dump 31GB to tape in 3 hours. We shall assume that a 200 inch/second 6250 bpi tape drive will have an effective speed of 200KB/second after accounting for operating system overhead, rewinding and tape mounts. Thus each drive can transfer .72GB/hour. Hence, we need 15 tape drives and 8 controllers. The address file could be written onto the batch disk drives and dumped to tape later. Obviously if we have 10 separate sites we will need 10 controllers and 20 tape drives (probably more due to unequal loading). One spare controller and drive would be desirable at each site.

IBM 3420-8 tape drives (6250 bpi, 200ips) cost about $30,000 apiece, and maintenance comes to $240/month each. IBM 3803-2 tape controllers, with 2 channel switches, and 4x16 tape switches run about $55,000 each. Maintenance comes to $265/month each. Thus:

Cost of Tape Drives, etc.
(one central site)

Item	Number	Total Cost Purchase $ cost	Total Cost Maintenance $cost/mo
3420-8 Tape drives	16	480K	3.84K
3803-2 Controllers	9	495K	2.39K

Note: we have included one spare tape drive and controller.

Cost of Tape Drives, etc.
(10 regional sites)

Item	Number	Total Cost Purchase $ cost	Total Cost Maintenance $cost/mo
3420-8 Tape drives	30	900K	7.20K
3803-2 Controllers	20	1100K	5.30K

Note: We are assuming 2 controllers and 3 drives/site so we can do tape merging. No spares. But we can back up the data base using one controller and 2 drives.

15.5. Front End Processors

Medium size IBM 3704 front end communications processors cost about $33,000 (to buy) and maintenance runs about $360/month. Also there is a cost per line of about $3600 (plus $25/month maintenance). We can connect 16 9.6KB/second full duplex synchronous lines to a 3704. Modem costs are not included here, but are discussed in the chapter "Communications".

If we have both RCIPs and CIPs online we have the following communications requirements at an average regional site.

Communications requirements
at an average regional site
(Assuming 4800 baud lines to concentrators)

Scenario	Lines To Terminal Concentrators	Lines To other sites
Worst Case Startup	44	4
Best Case Startup	18	4
Worst Case Normal	21	4
Best Case Normal	9	4

Communications requirements
at an average regional site
(4800 baud lines to concentrators)

Scenario	(without spares) 3704's	(without spares) Total Lines	(with spares) 3704's	(with spares) Total Lines
Worst Case Startup	3	48	4	64
Best Case Startup	2	22	3	38
Worst Case Normal	2	25	3	41
Best Case Normal	1	13	2	26

Note: We have assumed at each site 1 spare 3704 and the minimum of 16 or the actual number of line interfaces.

Thus our costs at an average site would run:

Communications costs
at an average regional site
(with spares)
(4800 baud lines to concentrators)

Scenario	Purchase $ cost	Maintenance $cost/mo
Worst Case Startup	362K	3.04K
Best Case Startup	236K	2.03K
Worst Case Normal	247K	2.11K
Best Case Normal	160K	1.37K

Note: We have assumed at each site 1 spare 3704 and the minimum of 16 or the actual number of line interfaces.

Thus our costs for all 10 regional sites would run:

Communications costs
at 10 regional sites
(with spares)
(4800 baud lines to concentrators)

Scenario	Purchase $ cost	Maintenance $cost/mo
Worst Case Startup	3.62M	30.4K
Best Case Startup	2.36M	20.3K
Worst Case Normal	2.47M	21.1K
Best Case Normal	1.60M	13.7K

Note: We have assumed at each site 1 spare 3704 with the minimum of 16 or the actual numer of line interfaces.

Alternatively, if 9600 baud lines are run to concentrators, we have the following:

Communications requirements
at an average regional site
(Assuming 9600 baud lines to concentrators)

Scenario	Lines To Terminal Concentrators	Lines To other sites
Worst Case Startup	33	4
Best Case Startup	13	4
Worst Case Normal	16	4
Best Case Normal	7	4

Communications requirements
at an average regional site
(9600 baud lines to concentrators)

Scenario	(without spares) 3704's	(without spares) Total Lines	(with spares) 3704's	(with spares) Total Lines
Worst Case Startup	3	37	4	53
Best Case Startup	2	17	3	33
Worst Case Normal	2	20	3	36
Best Case Normal	1	11	2	22

Note: We have assumed at each site 1 spare 3704 and the minimum of 16 or the actual number of line interfaces.

Thus our costs at an average site would run:

Communications costs
at an average regional site
(with spares)
(9600 baud lines to concentrators)

Scenario	Purchase $ cost	Maintenance $cost/mo
Worst Case Startup	323K	2.77K
Best Case Startup	218K	1.91K
Worst Case Normal	229K	1.98K
Best Case Normal	145K	1.27K

Note: We have assume at each site 1 spare 3704 and the minimum of 16 or the actual number of line interfaces.

Thus our total communication costs would run:

Communications costs
at 10 regional sites
(with spares)
(9600 baud lines to concentrators)

Scenario	Purchase $ cost	Maintenance $cost/mo
Worst Case Startup	3.23M	27.7K
Best Case Startup	2.18M	19.1K
Worst Case Normal	2.29M	19.8K
Best Case Normal	1.45M	12.7K

Note: We have assumed at each site 1 spare 3704 with the minimum of 16 or the actual numer of line interfaces.

16. Software

16.1. Operating Systems

On IBM (or plug compatible) systems we have 2 major alternatives:

(1) Run the standard operating system (CICS/MVS)

(2) Run the transaction processing operating system (ACP)

16.1.1. CICS/MVS

We could run the stock operating system, in this case CICS on top of MVS. A system of this type would provide a fairly congenial environment in which to develop the application programs. We could use various standard high level languages and database management systems. CICS supports distributed processing. However, such a system would require 8 to 10 times as many CPU cycles as the second alternative.

16.1.2. ACP

We could run a transaction processing operation system, ACP (Airline Control Program), which was originally developed for airline reservation systems. It is, however, a fairly crude operating system, supporting very few types of terminals and other devices. Its file system is very simple; only direct access files (and tapes) with record sizes of 381 or 1055 bytes are supported. It has a completely different interface than the standard IBM operating system (OS/VS), hence none of the regular OS compatible software products will run under this operating system. There are no IBM supported high level programming languages or database management systems available. There is a high level language (similar to PL/1) developed by Eastern and American Airlines which has been interfaced to ACP. We would urge its use (rather than assembly language) for writing the applications software.

Applications typically run 8 to 10 times faster under this operating system. However, applications program development typically requires 3 times as much effort.

16.1.3. Combination It is also possible to run one copy of each operating system under a virtual machine monitor known as "Hypervisor" at a loss of about 10% of ACP performance. The Hypervisor will not permit channel sharing. Nor can either operating system read the other's disk files (because of format differences) unless special purpose i/o drivers are written.

16.2. Applications Software

16.2.1. Cost per Man Year of Programming Effort

We expect that one man year of professional programmer's time will cost $90K when purchased from a software house. This figure includes all overhead: secretaries, supplies, management, buildings, etc. The Oak Ridge report estimated $50K per man year.

16.2.2. Size of Effort

We discussed the design and implementation of an electronic funds transfer system written in assembly language under ACP with IBM staff who worked on the project. That project required 30 man years of programming effort over a period of 18-20 months.

We believe that CIP functions are somewhat simpler and have therefore estimated that they will require 20 man years of programming effort over about 12-15 months.

Most of the remainder of the programming arises from RCIP transactions. Since we expect to do online VIN checking and zip coding, we believe this to be several times more complicated than the CIP functions and have estimated the effort at 100 man years of programming effort over 2 to 3 years.

Thus we have:

Applications Software Cost
(under ACP)

Function	Man Years	Dollars ($90K Man Year)	Dollars ($50K Man Year)
CIP transactions	20	1.8M	1M
RCIP transactions, etc.	100	9.0M	5M
Total	120	10.8M	6M

Applications Software Cost
(under CICS/MVS)

Function	Man Years	Dollars ($90K Man Years)	Dollars ($50K Man Years)
CIP transactions	7	.63M	.35M
RCIP transactions, etc.	35	3.2M	1.75M
Total	42	3.8M	2.1M

16.2.3. VIN checking and coding

R.L. Polk has a program called VINA which checks VIN's for correct format and which can decode the VIN to give vehicle type, fuel type, weight, etc. The program is presently written in IBM 370 assembly language and runs under IBM's standard operating systems (not ACP). It has been estimated to require about 300KB of main memory for this application (including tables)[59].

The program would have to be recoded to run under ACP. This should be straightforward and the effort is included in the estimates for RCIP transaction software costs. The database can probably be used with minimal

[59] The size of the tables depends on which model years and vehicle characteristics one is interested in.

modifications.

Polk charges $9,000 for the basic program. There is a $600 per model year charge for the database used for checking VIN's. For decoding vehicle characteristics (for model year 1966 and later) the charges vary depending on which vehicle characteristics are to be decoded. Charges run $275 to $1500 per model year for autos for items which might be of interest for gas rationing[60]. Truck charges run $620 to $2000 per model year[61].

The code requires only a very modest amount of CPU time.

16.2.4. Zip coding, address corections

Polk also has a program called ORACLE which finds zip codes of addresses and does some spelling correction of addresses. This program is also in IBM 370 assembly language and runs under standard IBM operating systems. The program is used for batch processing and relies on sorting the address file to operate efficiently.

While the program might be suitable for the initial creation of the NVRF, it would not be suitable for online zip coding and address correction during RCIP transactions. We would expect then to purchase the database used by the program (and possibly the spelling correction algorithm). The entire program can be leased for $12.5K initially plus $2.8K per year. Standard database maintenance is done only once a year. More frequent updates cost $250 each. It is not clear whether these charges are per agency, site or machine. The lease of the database (and spelling algorithm) should cost no more than leasing the program and database.

A new program would have to be written using either Polk's spelling correction algorithm, or some other, to run under ACP in an interactive (i.e., unsorted) environment.

The Post Office has a similar program written in COBOL. It is discussed in the Oak Ridge report.

In calculating hardware requirements we have assumed that all city names would be stored in main memory and that only a single disk seek would be required to recover the the zip code from the city and street address. We have not included processor time for spelling correction (which is considerable) because we doubt that it is practical in an online environment[62].

---

[60]The smaller amount only buys the model and fuel type.

[61]The smaller amount buys chassis/body type, fuel type, and manufacturing gross vehicle weight.

[62]Also additional seeks (and buffers) would be required.

17. Privacy

17.1. The Problem

In order for the proposed gasoline rationing system to work we must collect personal information on most of the adult population of the United States, specifically address and vehicle ownership. For audit purposes we must also record the time, place, and participants of every transaction involving gas rationing, e.g., coupon issuance.

It is clear from the Federal Privacy Act, and other Congressional debates that protecting the privacy of individuals is an important consideration in the design of federal data processing systems. While it is essential to the proper operation of a gas rationing system that certain types of personal information be recorded it seems clear that:

(1) That amount of personal information should be minimized, and

(2) Steps should be taken prevent its disclosure to other parties or for uses unrelated to gas rationing.

Aside from a generic congressional mandate to maximize individual privacy, it seems likely that adequate protections for personal privacy will be important in securing Congressional and public acceptance of any gas rationing plan.

Some of the design issues involving privacy are discussed below.

17.2. RCIP Transactions

RCIP transactions involve the transmission of personal information. Thus it would be desirable if these transmissions were encrytped. Encryption is also necessary to prevent linetappers from entering fake records into the database.

17.3. CIP Transactions

CIP transactions involve very little personal information (VIN, license plate number). We do not feel that they need to be encrypted. To do so would add perhaps $100-200 to the cost of each terminal (at present).

17.4. No indexing on owner name

Access to the NVRF is never necessary on the basis of the vehicle's owner name. We would therefore suggest that no such indices be permitted. This would make the database somewhat less useful to peepers.

17.5. Design of transactions

To the extent possible transactions should be designed so that the clerk enters personal information and the computer system merely confirms that it is correct (rather than permitting inquiries by the clerk).

Furthermore, personal information should never be used when random numbers would suffice. For example, one way for the computer to acknowledge a CIP transaction (assuming that the clerk has entered the check number and VIN) would be to return the licence plate number. However, we could just as easily print a couple of random digits on the check and send them to the clerk as an acknowledgement. The additional expense would be minor.

17.6. Isolation of Regional Computer Centers

If the computer system is physically distributed (into 10 regional centers for example) and if we limit the type of information which can be sent between computer centers, we might achieve some increase in the cost expended by a peeper to gain access to all possible records. Specifically, it does not appear necessary that vehicle transfers between regions (either by moving or sale) require that the new region tell the old region anything about who owns the vehicle in the new region.

17.7. Encrypting the Database

At present it would be quite expensive to encrypt the database, especially if we attempt to adhere to the Federal Data Encryption Standard[63]. Sometime in the next few years it should be possible to purchase disk and tape controllers which will encrypt the data efficiently. Until such time we are inclined to suggest that the database not be encrypted.

In any case we believe that fraudulent RCIP transactions are probably the easiest and most likely method of peeping at records. Encrypting the database will not stop these attacks.

If encryption is desirable we would suggest that individual records be encrypted separately (not entire sectors), that many keys be used[64], and that alternatives to DES be considered.

17.8. Legal Peeping

Various law enforcement and intelligence agencies were exempted from the 1974 Federal Privacy Act. If such agencies are to be denied access to the gas rationing files (except for fraud enforcement purposes), then special legislation, perhaps similar to laws protecting census files, would be needed.

17.9. Is the Federal Data Encryption Standard Secure?

There has been considerable debate as to whether the Federal Data Encryption Standard is secure against serious attack. We expect that affluent peepers could break the cipher by exhaustive search for the key sometime in the mid 1980's. Triple encryption (with 3 separate keys) may be of some help here. So will the use of a multiplicity of keys - this reduces the value of breaking a single key.

---

63 The DES seems to have been designed to be computed with special hardware. On a large conventional computer it requires some 400 instructions per byte encrypted. Decryption takes the same mount of time.

64 One could algorithmically generate a unique key for each record by combining the VIN with a master key.

18. Summary

Online distributed gasoline ration check issuance and reconciliation systems are technically feasible. They can stop several types of fraud. Such systems however would require several years to implement. Cost estimates vary widely depending on what schedule for coupon distribution is assumed.

18.1. Cost Structure

For teller based systems it is clear that labor costs will dwarf all other items. Communications costs will come in second. Terminals come in third while computers and storage trail well behind.

Costs
Both CIPs and RCIPs Online

Item	Capital Outlays		Quarterly Outlays (additional)	
	Low	High	Low	High
Teller Labor	0	0	$100M	$200M
Communications	$2M	$20M	$5M	$38M
Terminals	$27M	$275M	$800K	$6M
Computers	$3.3M	$22M	$60K	$300K
Storage	$11M	$18M	$190K	$310K
Software	$6M	$11M	$100K	$300K

We doubt that teller labor costs will be greatly affected by whether or not the system is automated. However, there is considerable uncertainty in the estimates of teller transaction processing times. Labor costs are also sensitive to peak/average traffic ratios, the viability of variable staffing (by part time employees) to meet peak traffic periods, and non-uniform population distribution which will require additional CIPs and RCIPs to serve sparsely populated areas. RCIP labor costs are fairly sensitive to the error rate in the NVRF, the magnitude of which we are uncertain.

Similarly the number of terminals required will vary with the teller transaction times, peak/average traffic ratio, nonuniform population distribution and NVRF error rates[65]. The number of terminals will also depend on the schedule adopted for coupon distribution. Communications costs are mainly for lines between terminals and concentrators and thus they are very sensitive to the number of terminals and their geographic distribution.

Computer costs are largely a function of peak traffic rates, which in turn depend on the schedule, peak/average traffic ratios, and NVRF error rates.

[65] NVRF error rates primarily affect the number of RCIP terminals required.

Storage costs depend on whether or not the disk files are to be duplicated. For low traffic (normal scenarios) storage system costs depend only on file size. For very high traffic scenarios storage system costs depend only on peak traffic rates.

18.2. Automatic Teller Machines

Each automatic teller machine (at $15K plus installation) could operate 16 hours/day, replacing two human tellers. Thus automatic teller machines are cheaper than humans if gas rationing is to be enforced for substantial periods of time. However, the enormous capital costs[66] entailed would only be warranted if DOE were fairly certain that gas rationing would be adopted. We expect several years would be required to construct and install a complete complement of these machines.

18.3. Computer Decisions

We recommend that DOE acquire computers with at least 32 bit word size, to minimize software problems. Furthermore we would recommend the use of a transaction operating system such as IBM's ACP. Hashing is clearly the preferred access method for organizing the files. Because of the modest incremental cost and the substantial impact of system outages we would recommend that the disk systems be duplicated and that some spare CPU capacity be acquired.

18.4. Physically Distributed Systems

For heavy traffic scenarios, it does not appear to make a great deal of difference in cost whether one places the computers in 10 regional processing centers or in one central processing site. This decision should probably be made on other than economic grounds (administrative convenience, privacy considerations, political preferences)

For low traffic scenarios[67] a single site will be cheaper than 10 regional sites, but the cost difference is very small compared to the teller costs.

18.5. Online RCIPs

We believe that online RCIPs are worthwhile. For a modest increment in cost we believe DOE will obtain significant improvements in the accuracy of entries to the database and in providing timely responses to citizen queries. In conjunction with online CIPs, online RCIPs can be used to stop certain types of fraud.

18.6. Online CIPs

The desirability of online CIPs hinges on estimates of likelihood of going to gas rationing, the expected duration of rationing, and anticipated fraud rates. Major capital commitments for terminals would have to made well in advance of system operation. If fast startup is desired, communication lines would have to be installed ahead of time and paid for during standby

---

[66] $250M to $750M purchase cost.

[67] Such as the best case normal scenario.

mode.

If one is optimistic about fraud rates[68], thinks that rationing will probably not be needed, and if needed would last only a few months, yet wants a system which could be placed in operation very quickly (thereby requiring that communications lines be leased during standby periods), then online CIPs will not be cost effective.

If one is pessimistic about fraud rates and thinks rationing likely to be adopted soon and to last several quarters or more then online CIPs will pay for themselves several times over.

18.7. Implementation Time

Even under very optimistic assumptions about procurement times, these online system would require at least 2 years to implement.

[68] Estimates of recovery costs range from $35M to $130M per quarter.

TECHNICAL INFORMATION DEPARTMENT
LAWRENCE BERKELEY LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720

TECHNICAL INFORMATION DEPARTMENT
LAWRENCE BERKELEY LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720