~~TOP SECRET~~

NATIONAL SECURITY AGENCY
FORT GEORGE G. MEADE, MARYLAND

CRYPTOLOG

MAY 1978

P.L. 86-36

~~THIS DOCUMENT CONTAINS CODEWORD MATERIAL~~

~~TOP SECRET~~

~~Classified by DIRNSA/CHCSS (NSA/CSSM 123-2)~~
~~Exempt from GDS, EO 11652, Category 2~~
~~Declassify Upon Notification by the Originator~~

CRYPTOLOG

Published Monthly by P1, Techniques and Standards,
for the Personnel of Operations

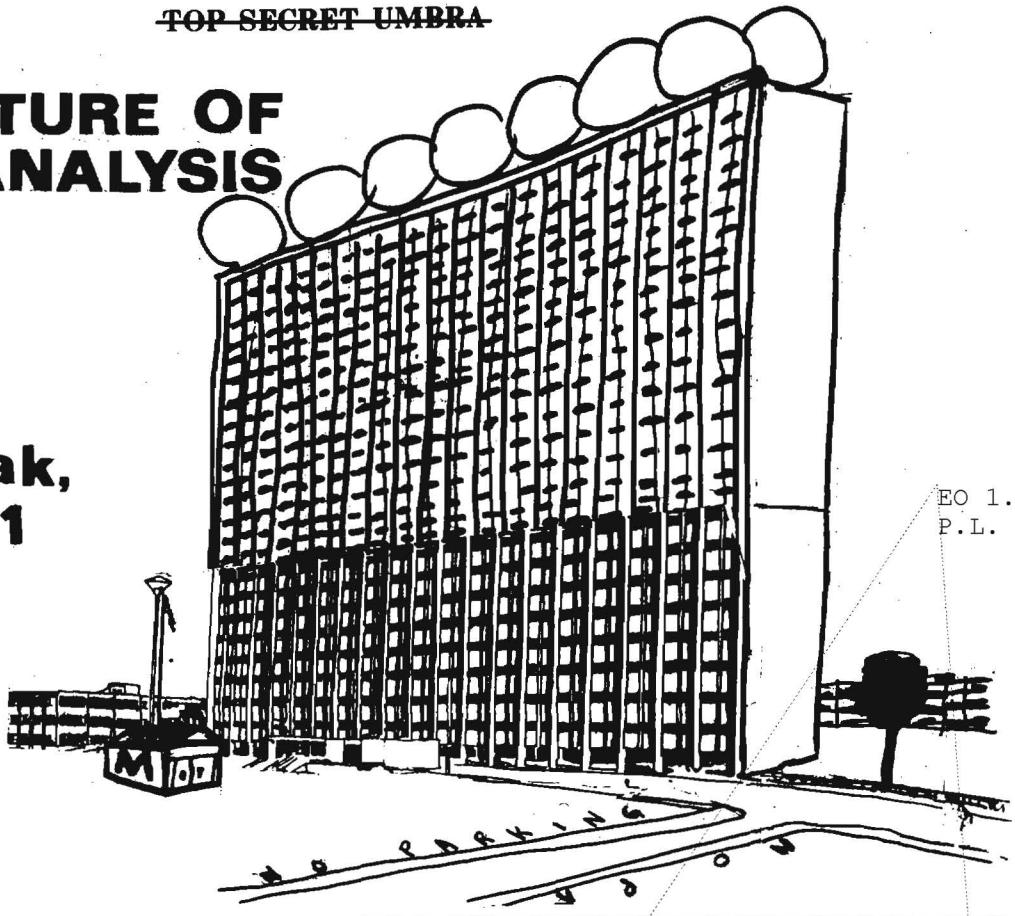
VOL. V, No. 5 MAY 1978

PUBLISHER WILLIAM LUTWINIAK

For individual subscriptions
send
name and organizational designator
to: CRYPTOLOG, P1

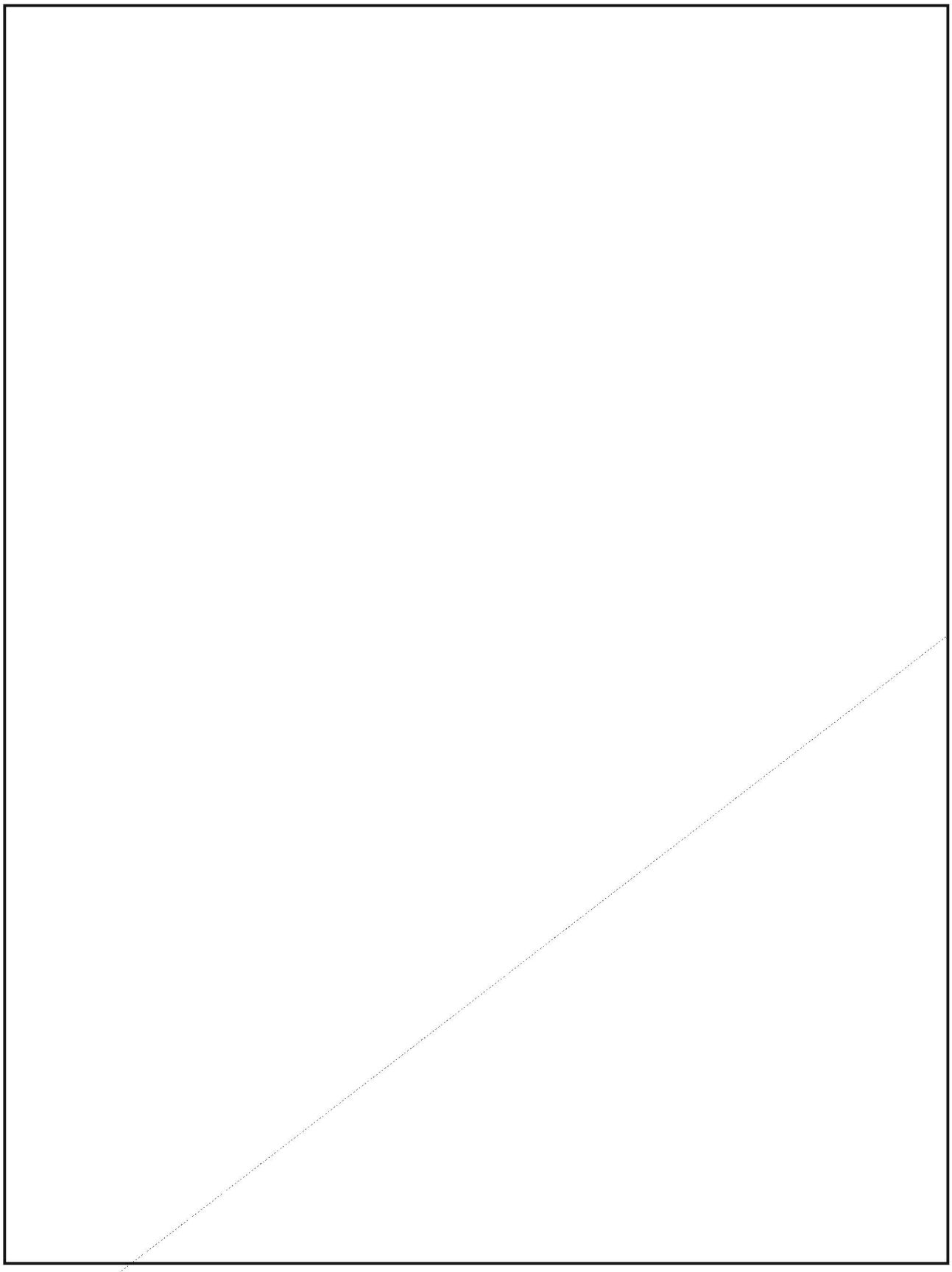
THE FUTURE OF CRYPTANALYSIS

William Lutwiniak, Chief, P1

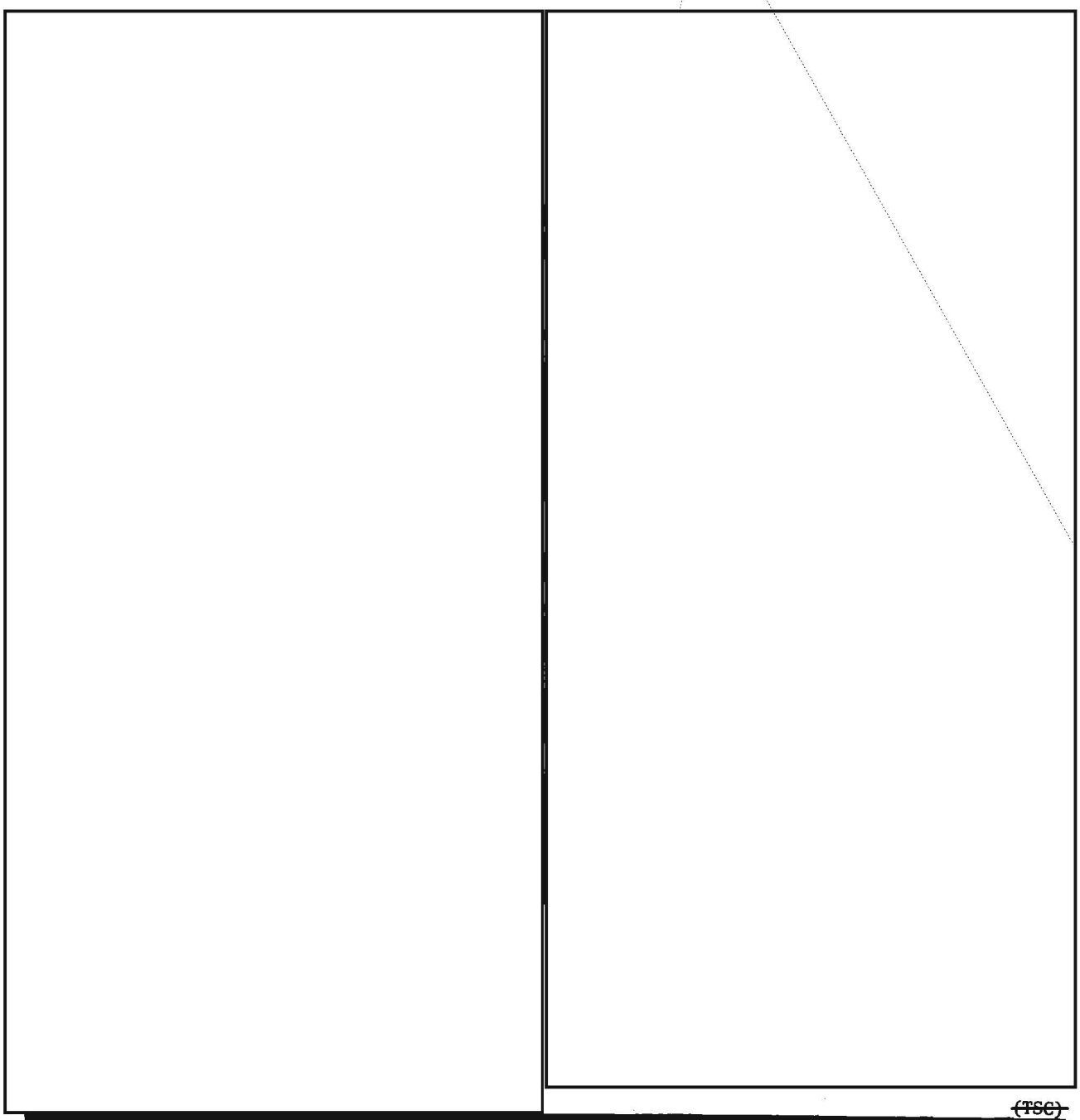


The second most frequent question I'm asked (the first concerns promotions) is "Does the cryptanalyst *have* a future?" Predicting is a risky business. Did you happen to catch, in all the school closures and cancellations announced on the radio on that snowy Friday last week, the postponement of the meeting of the Clairvoyants Society?. . .

Complete text of the keynote address delivered by Mr. Lutwiniak on 24 January 1978 to the third annual seminar-workshop series "Cryptanalysis: Contemporary Issues." The series is offered as a course (CA-305) by the Cryptanalysis Division of the National Cryptologic School. Other papers presented during the 1978 series will appear in future issues of CRYPTOLOG.



EO 1.4.(c)
P.L. 86-36



~~(TSC)~~

Do you want to attend next year's seminar?

The seminar-workshop "Cryptanalysis: Contemporary Issues" is offered every winter by the Cryptanalysis Division of the National Cryptologic School. It is designed to bring SIGINT and COMSEC cryptanalysts, cryptologic mathematicians, and other interested persons up to date in the status and trends in the current practice of cryptanalysis. Speakers from various parts of the Agency make presentations covering cryptanalysis and related fields.

Usually the seminar lasts three days. It consists of the keynote address (delivered in the Friedman Auditorium) and 15 smaller presentations. Each presentation is given twice. To receive credit, participants must attend the keynote address and five presentations.

The seminar, which changes every year, attracts participants from throughout the Agency. If you are interested in attending the 1979 seminar, look for the course announcement in December 1978. If you haven't seen it by Christmas time, check with your training coordinator. Distribution may have slipped up. J.E.D. (U)

CALLSIGNS AND WARC 79 [redacted] P13

747 Table of Allocation of International [redacted]

The Federal Communications Commission (FCC) has proposed a significant change in radio callsigns for WARC 79 (World Administrative Radio Conference, Geneva, 1979), namely, that callsigns be *unique* and capable of fully *automatic* monitoring. FCC also proposes that such unique callsigns be applied to every transmitter or transmission "which *could* propagate" (italics added) beyond the boundaries of the country to which they belong. The FCC also wants to drop from the radio regulations the exemption for military transmitters. This scheme, if it is presented as the U.S. position at WARC 79, could have a marked effect on SIGINT and COMSEC of many countries, and hence deserves notice.

The unique callsign scheme was presented by the FCC in its Fifth Notice of Inquiry (NOI) for WARC 79, published in *Federal Register*, 31 May 1977. The specific language appears in Appendix 3 as proposed changes to the *Radio Regulations* of the International Telecommunication Union (ITU) and as Resolution G-C.

Resolution G-C, "Relating to Automatic Identification," states, in part:

> "The General World Administrative Radio Conference, Geneva, 1979, considering
>
> a) the state of the art in respect to identification
> b) the need for unique identification
> c) the possibility of inadvertent operator error
> d) the ever increasing number of active transmitters not only within existing administrations but also noting the ongoing assignments of new call sign and selective calling systems
> e) the economics of presently available equipment which is directly applicable to automatic identification
> f) the possibilities for use of automatic, faster message transmission service in conjunction with automatic identification equipment
> g) the increased ease of resolving cases of harmful interference and of ensuring compliance with the provisions of the Convention and the Radio Regulations
>
> resolves
>
> 1. that administrations shall implement automatic identification as provided in Article 19 MOD [= *mod*ification -- see below] *at the earliest possible time*, and
> 2. that automatic means of identification should be adopted by *all* administrations." (Italics added.)

The FCC then proposes substantial changes to *Radio Regulations* Article 19, "Identification of Stations." It proposes replacing the current (1976) No. 735 with MOD 735, as follows:

Current 735 "Transmissions without identification or with false identifications are prohibited."

MOD 735 "Transmissions and *transmitting stations* shall be *uniquely identified*. Administrations shall make every effort at the earliest possible time to introduce and use automatic identification. On frequencies assigned for international use, means recommended by CCIR [International Radio Consultative Committee, of the ITU] shall be utilized." (Italics added.)

The FCC then proposes the suppression of the current Nos. 736 and 737A of the *Radio Regulations*, in order to remove exceptions to the modified No. 735. No. 736 currently provides an exemption for survival craft and emergency radio beacons; the FCC would eliminate this exemption in favor of callsigns that satisfy MOD 735. No. 737A currently provides an exemption for some space stations (e.g. satellites), which exemption would also be eliminated. No. 737A specifies various acceptable kinds of callsigns, station identifications, or selective call numbers -- which the FCC would accept.

No. 738 would remain unchanged. It specifies regular identification signals, at least hourly, and ends with the statement that ". . .administrations are *urged* to ensure that wherever practicable, superimposed identification methods be employed in accordance with CCIR Recommendations." (Italics added.)

Nos. 739, 740, and 741 would be suppressed. No. 739 states that the identifying signal shall be transmitted by methods which "do *not* require the use of special terminal equipment for reception." (Italics added.) Nos. 740 and 741 also conflict with the automatic monitoring and uniqueness requirements.

The FCC gives as its reason for suppression of Nos. 736 and 737A, "To provide universal, unique, and automated identification." Its reason for suppression of Nos. 739, 740, and 741 is "Consequential to above proposals."

The FCC then proposes that No. 742, which allows each ITU member to establish "its own measures for identifying its stations used for national defence," be suppressed. The FCC gives as its reason for this proposed suppression, "Unnecessary to incorporate Convention

provisions in the Radio Regulations" (that is, the general authority for national defense stations is given in Article 38 of the Convention). However, the deletion of No. 742 is clearly intended to encourage and facilitate the unique identification of military transmitters and transmissions. The treatment of No. 742 at WARC 79 could produce interesting alignments of countries.

The clearly stated aim of all this would be to allow automatic monitoring of all radio stations by causing them to transmit unique callsigns or identification numbers in an automatic and standardized way. The adoption of this regulation would affect millions of stations -- fixed and mobile, CB, radio-amateur, and earth satellite -- which currently are not required (47CFR 25.206) to identify themselves in the United States and in 150 other countries. It would completely change the economics and practical aspects of radio monitoring, interference notifications, and regulatory enforcement in developed countries where there are numerous radio transmitters. It would also have a marked effect in the ITU allocations and statistical studies of the HF spectrum. Although millions of transmitters which already use callsigns would be affected, the FCC proposal does not stop there.

"Harmful Interference" vs. Capability to "Propagate Internationally"

In Section II, "Allocation of International Series, and Assignment of Call Signs," of Article 19, the FCC further proposes a significant regulatory change which would markedly increase the number of stations to be assigned unique international callsigns. No. 743 now reads,

> "All stations open to the international public correspondence service, all amateur stations, and other stations which are capable of causing *harmful interference* beyond the boundaries of the country to which they belong, shall have call signs from the international series allocated to each country as given in the Table of Allocation of Call Sign Series in No. 747." (Italics added.)

Here the deciding criterion of "other stations" is that of "harmful interference," which is defined in Annex 2 of the 1973 ITC (TIAS 8572) as

> "any emission, radiation or induction which *endangers* the functioning of a radionavigation service or of other safety services, or *seriously degrades*, obstructs or repeatedly interrupts a radio communication service operating in accordance with the Radio Regulations." (Italics added.)

This is a stringent criterion which requires the occurrence of sustained severe interference. Hence most stations are not required to use international callsigns, and the burden falls on the victim to prove the harm and identify the station exerting the "harmful interference."

The modified language the FCC proposes is quite different, namely:

> **MOD 743** "Each station whose signal *could propagate* internationally shall uniquely identify itself such as by a call sign formed pursuant to No. 747. Identification shall preferably be by automated means using the applicable Recommendations of the CCIR. (See Resolution G-C)." (Italics added.)

This is clearly a completely different criterion, for "harmful interference" does not have to manifestly occur. Instead, the criterion is merely the technical *capability* for detectable propagation across a national border, into the international ocean areas, or into international space, particularly the equatorial geostationary satellite orbit, which is becoming crowded.

The effect of MOD 743 would be to require that a much larger population of transmitters, all over the world, send unique identifications in a manner that could be automatically monitored. Tens of millions of transmitters, particularly mobile stations, would be affected by this.

Considered in a U.S. context, virtually all mobile and CB transmitters which *could* propagate across national borders would be affected. Earth satellite stations and even radio relay stations near borders would be affected. The modified language of 743 also requires that each station "shall uniquely identify itself," while the existing language only requires that it "shall have" a callsign. This imposes a requirement for automatic identification whenever a transmitter comes on the air, rather than at the operator's discretion, and the use of improvised or changing callsigns, as in CB, would not satisfy the new regulatory language. The FCC has been trying for some years to get automatic identification features into mobile and CB radio (e.g., Docket 2437) and this has been fought by the radio industry. But if WARC 79 adopts such a provision, then Regulation 47 USC 303(R) will make that new Radio Regulation applicable to the United States, and FCC-type approval can exclude all new transmitters which do not satisfy this requirement.

Considered in a foreign context, the adoption of such regulations not only will affect internal radio operations and monitoring, but will also produce a great increase in radio negotiations between countries and in the reports to the ITU, IFRB, and CCIR as countries attempt to reduce the interference their stations cause or suffer. Automatic monitoring, with computer analysis

and accounting, will lead to much more complete models of propagation. Morever, it will provide a great deal of data about radio traffic activity.

One other proposed FCC modification -- to Article 16, "Reports of Infringement," reads,

> "If an administration has information of an infringement of the Convention or Radio Regulations, committed by a station *over which it may exercise authority*, it shall ascertain the facts, fix the responsibility and take the necessary action." (Italics as in text.)

The FCC states that this change is "To permit action against possibly unauthorized stations committing infringements." This closes a legal loophole by which a country could ignore violations by a transmitter in its territory by claiming it had *not authorized* the station.

Aggregate Effect of FCC Proposals

The aggregate effect of the modifications to Articles 16 and 19 and of Resolution G-C would be to establish a framework for global automatic monitoring and much tighter enforcement of radio laws and regulations. If these proposals, or something like them, are adopted at WARC 79, the global radio spectrum will be a much more tightly managed resource, and international engineering of radio systems will become a standard phenomenon, rather than a rare one. The Soviet delegate to CCIR, Sviridenko, favors such central planning, management, and engineering of the radio spectrum, using computer propagation models (largely developed by the United States). At present the "priority rights" of current spectrum users, especially in HF and space systems, are defended by the industrial nations because radical changes in allocations and radio links would produce unpredictable effects. With global automatic monitoring made economical and practical, much more intensive use of the spectrum could be undertaken and propagation effects predicted. Interference, or any infringements, could be quickly identified, and corrections demanded under Article 16. All this would produce a great deal of change in radio usage and data about radio traffic over the next 25 years, particularly in the congested regions of the spectrum.

Effect on U.S. and Foreign COMINT and COMSEC

The unresolved questions is how this will affect U.S. SIGINT and COMSEC, and corresponding foreign SIGINT and COMSEC.

HISTORICAL NOTE ON MILITARY CALLSIGNS

J. A. Meyer

Since the first international radio treaty in 1906, nations have always reserved complete freedom for their military and naval stations, opposing international regulation except with regard to distress messages and interference. This is expressed in Article 38 of the 1973 International Telecommunications Convention (TIAS 8572). The United States has always maintained this same reservation. Section 303(o) of the Communications Act of 1934 states that the FCC shall "have authority to designate call letters of *all* stations" (italics added). Section 305(a) of the same Act provides an exemption for government-owned stations, as follows:

> "Radio stations belonging to and operated by the United States shall not be subject to the provisions of sections 301 and 303 of this Act."

But then the exemption is reduced for *callsigns* by Section 305(c), which states:

> "All stations owned and operated by the United States, except mobile stations of the Army of the United States, and all other stations on land and sea, shall have special call letters designated by the [Federal Communications] Commission.
>
> (Note that the word "except" pertains only to "mobile stations of the Army of the United States.")

Therefore the removal of No. 742 would reinforce the authority of the FCC to assign callsigns to U.S. military stations under 47 USC 303(c). Even the basic concept of freedom for national defense stations, as declared in Article 38 of the ITC was challenged by the USSR in 1932 when that government was first invited to the ITC in Madrid. The Russians proposed that military stations be regulated by the same rules as nonmilitary stations, and although that proposal was defeated, the challenge might be resurrected at WARC 79, where the USSR could expect greater political support than in 1932. If No. 742 is deleted at WARC 79, as the FCC proposes, a considerable effort by U.S. COMSEC to get U.S. military stations to use changing callsigns would be undermined. Hence the FCC proposal would affect U.S. security unless 47 USC 305(c) is amended to compensate for this.

~~(FOUO)~~

~~SECRET~~

[redacted]

The FCC, in issuing these proposals on 31 May 1977, invited comments. At that time the Office of Telecommunications Policy (OTP), a White House staff group established in 1970, was the official organ for coordinating and presenting the comments of the government to the FCC. The deadline for the comments on this fifth NOI has passed, and the next NOI is expected in early 1978. The OTP is being disestablished, and it is not clear where the coordination function for NOI responses will eventually land. The Department of Defense, as the largest user of international telecommunications in the world, has a special status and in 1973 was a member of the U.S. delegation to the Plenipotentiary Conference of the ITU. Other government departments respond directly to the NOIs, and the Department of Defense could also reply directly, particularly where national security considerations apply.

The FCC proposal for unique, fixed, automatically recognizable callsigns, having been presented as the public position of the U.S. government in the *Federal Register*, will be widely read around the world and may be introduced and supported by other countries at WARC-79, whether or not the United States presents it.

The FCC proposal represents a consolidation of ideas which have been emerging and finding application -- especially in aeronautical mobile and maritime mobile radio communications -- for some years. Selective calling systems and automatic monitoring equipment, as the FCC notes, are already in service and have proved their value. These facts will be noted at WARC 79. The FCC proposal is a logical generalization of existing practices and a recognition that automatic monitoring -- and the automatic identification of practically *all* transmitters -- is an essential condition for radio planning and management for the next 20 years and beyond.

Effect on Agency's Mission

[redacted]

~~(C - CCO)~~

T.A. IMPLICATIONS OF F.C.C. PROPOSAL

P14

[redacted]

~~(S - CCO)~~

~~SECRET~~ ~~HANDLE VIA COMINT CHANNELS ONLY~~

~~SECRET~~

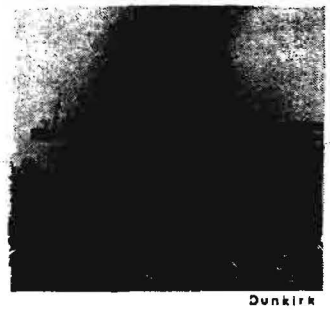
[redacted]

~~(S — GCO)~~

Solution to NSA-crostic No. 13

By [redacted]

(CRYPTOLOG, April 1978)



Dunkirk

P. [William] Filby, "ULTRA [Was] Secret Weapon [That Helped Defeat Nazis]," CRYPTOLOG, December 1975 (U).

"Unhappily, it was not unusual for holders of the German [decrypts] to have to forgo using them for fear of compromising the cypher break. One such occasion was the bombing of poor Coventry; enemy plans were known beforehand, but to defend the city would have aroused German suspicions."

(U)

Answer to "Telephone Problem II"

By [redacted]

(CRYPTOLOG, April 1978)

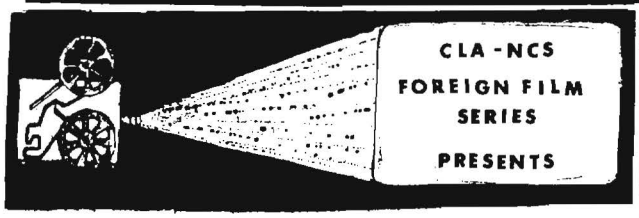
P.L. 86-36

The upper right-hand corner of the covername allocation is:

OVAL	ROWBOAT	-	ADAGE	-	FIGHTER
ALBUM	COMPRESS	-	LACEWING	-	SCHOOLBOY
	PESO	SEEDLING	TONIC	APEMAN	COUNTRY
			GERUND	LOUVER	PLODDER
				TYPHOON	BARRETTE
					MESA

Once you recover the method of generation, can you deduce the *source* of the covernames?

(U)



The Crypto-Linguistics Association and the National Cryptologic School will present

in May: "The Shop on Main Street" (in Czech, with English subtitles)

Friday, 5 May, 0930

in June: "Heroes of Shipka" (in Russian, with English subtitles)

Thursday, 8 June, 0930

Both films will be shown in the Friedman Auditorium. All are welcome.

Announcements with details about the films will be mailed to CLA members and will be posted throughout the Agency. Look for them. We'll see you in the Auditorium! B.Y.O.P.C.

(U)

PROJECT UTENSIL:
The DDO Data Dictionary/Directory

P13

What *is* a data dictionary/directory? Just as an ordinary dictionary contains information about words, a data dictionary contains information about data. It does not contain the actual data that forms the data files, but contains pertinent information about that data, its attributes, and relationships. There are many definitions in the commercial world, but what the dictionary contains is information in two forms:

- the "what" information -- the *data dictionary* (description of the data elements); and
- the "where" information -- the *data directory* (the location and use of the data elements and their relationships to other data elements, records, files, procedures, etc.).

Why Combine the Two?

By combining the dictionary and the directory into a single "data dictionary/directory" we have the ability to provide:

- coordination and control in systems development;
- assistance in search for relevant data during design;
- a means of identifying and reducing data redundancy;
- an increased data-transfer capability between systems;
- data standardization;
- administrative support;
- documentation support;
- data-definition support.

Thus, the system provides a versatile tool for managing the DDO data resource.

Brief History of Data Dictionaries

Data dictionaries are not new, either in the commercial world or within the government. Their existence is related to, and probably can be entirely attributed to, the Data Base Management Systems (DBMS) and many are an integral part of a DBMS. The National Bureau of Standards has published an extensive report on seven of the commercial dictionaries and eleven government-agency systems.

Slightly edited version of a talk given in 1977 at two meetings of CISI's Special Interest Group on Information Processing Systems (SIG/IPS).

The National Bureau of Standards also sponsored a Federal Information Processing Standards (FIPS) task group -- TG-17 -- specifically to address guidelines for establishing data dictionary/directory systems. That task group published its report in late 1977.

NSA also has several data dictionary efforts in various stages of development. They are:

- TEDS, in W, which uses a data dictionary on the M-204 computer system that has been operational for some time;
- an effort in A to utilize the TEDS experience in developing a dictionary for STEPSTONE on the M-204;
- HOLLYHOCK, a project to support L, M, N, and E, which will use a data dictionary developed on the M-204 by T33;
- INLAND, a project to maintain continuity on R tasks, which uses a data dictionary developed in R;
- an IBM data dictionary that is being used on a system developed for a field site by T; and
- the most recent addition, the Cullinane Data Dictionary, which was purchased by T for use with the IDMS Data Base Management System, and which is currently being evaluated.

DDO's Need for a Data Dictionary/Directory

Why does DDO need a data dictionary/directory? If you have ever attempted to solve a problem that required you to find all the possible files, manual or machine, that might contain information about your particular subject, you know one reason why we need a data dictionary. For those of you who have never made that attempt, I have an example of such a situation. About 2 years ago I was asked to help locate all the data files containing geographic coordinates, grids, and/or other means of identifying a point on a map. The requester also wanted to know if there was any associated software to process the geographic information for selecting records by area. I found no means to locate either the files or the software without surveying each organization. How, then, do managers answer questions from auditors, customers, and the Director if there is no means to assure that they have all the information?

Other questions that a data dictionary/directory system might answer are:

- How many programs in the COBOL language are there on each machine?
- How many A files are resident on the IBM 370/168 system?
- Is there a standard for aircraft type?
- What files contain aircraft types associated with the Bulgarian or Hungarian air forces?
- How many B FORTRAN programs are there for the IBM 370/158 system?
- What computers are on the NSA Network (PLATFORM)?

Another illustration of a dictionary's use comes from [redacted] presentation on the G project, GEISHA, in which he discussed their current operations and the problems that G has encountered. He explained that one of the problems is the existence of many individual processes carrying out similar functions. Elimination of duplication and related ills requires a coordinated effort to create a single G system.

Think of DDO as being similar to what Mr. [redacted] reported about G processes. Many individual processes? Similar functions? Duplication? . . . The fact is that there is a system for A, a system for B, a system for G, a system for V, and a system for W. I do not propose that we design a single system for DDO as a whole, but a data dictionary containing descriptions of all the systems in common terms would eliminate many problems, or, at the very least, would help us to recognize problems where they exist.

Let's take a look at ourselves in DDO and diagnose the situation. The DDO organization contains several Groups with similar functions, such as TA and CA. This results in similar computer processes because the analysts' needs are similar. Therefore, it follows that data bases to support these functions will be nearly identical in structure. Well, similar or nearly identical, but not necessarily recognized as being such, because DDO is organized by *area*, and, since each is supported by different computer experts, the design of similar data bases would be different. Add to that the different terms used by analysts in each area and discipline, along with the use of acronyms and abbreviations, and it would be impossible to recognize the similiarities between data bases unless you carried out a thorough study. Therefore we need to agree upon common names and definitions for common fields of information.

Another situation is the result of the size of the Agency and the number of files required to support its functions. PLATFORM, a project to link computers, will make access from one computer to another a reality. I contend that having access to one or more machines doesn't give me any capabilities I don't already have, unless I have an inventory of what is available on those computers. Just having an inventory isn't enough, either, if you don't know how to identify or access a file. You also need to know more about the file if you plan to use it.

Take the following hypothetical example. For some research reason you would like to know the number of current NSA male employees with blond hair, born in New York City, who were hired by the Agency between 1956 and 1970. You want to be sure you have looked in all files which could possibly have information to answer your query. But you do not want to search files which wouldn't possibly have the proper information. With a data dictionary/directory data base containing pertinent information about all the available files, you could narrow your search to only those files which contain information about Agency employees, city of birth, date of hire, and color of hair. The system would then provide from the contents of the data dictionary/directory data base all the information available either on how to extract the information yourself or who to contact to get what you are looking for. This example isn't a typical DDO problem, but it enables me to add that the system would not allow you information about files if you do not have need-to-know or proper credentials. You would have, at minimum, identification of all files which could help you get your answer. P.L. 86-36

So the solution to our data-management problem is to to have a DDO Data Dictionary/ Directory. With that dictionary/directory we would no longer have the situation in which the same data, used in two data bases, would be described differently.

Yes, the DDO Data Dictionary/Directory can be the solution, but only if the contents and the data base are current and accurate and if they represent all facets of the DDO data resource. Descriptions of our data elements, data fields, records, files, and data bases are required to build the Dictionary/Directory data base. The following are two examples of descriptions of *data elements*:

DATA ELEMENTS

Name	Social security identification
Abbreviation	SSN
Synonym(s)	Social security account number
Definition	A unique indication of an individual and his Social Security account
Date approved	710701
NDSC identification	NI-0003
(Etc.)	

Name	Sex
Abbreviation	-
Synonym(s)	-
Definition	The division of human beings into groups based on physiological characteristics
Date approved	710225
NDSC identification	00056
(Etc.)	

The following is an example of a *record*, which can consist of one or more data elements:

RECORD

Name	Personnel record
Abbreviation	PERS RCD
Synonym(s)	-
Definition	The record of a specifi NSA employee
Field 1:	
Name of data element:	Social security number.
Length:	9
Configuration:	Numeric
Update authority:	M3
(Etc.)	
Field 2:	
Name of data element:	Sex
Length:	1
Configuration:	Alphabetic
Update authority	M3
(Etc.)	

What information, then, should we collect for inclusion in the Dictionary/Directory data base? What are the data elements? What are their defintiions. Where are the data elements used? The document "Data Standards for SIGINT Activities," promulgated as Annex A of USSID 414, is the only centrally documented source of data elements with definitions. A Computer Record Format File documents some computer jobs with field names. Except for the published standard Data Elements, these field names -- which are arbitrarily assigned -- are subject to the problems referred to earlier: different terminology and ideas about similar fields. Therefore, we have different fields represented in several files under the same name, and the same field represented in several fields under different names. To use everyday examples, assume that one file contains information on "STOCK" (in the sense of "livestock," with data pertaining to sheep, cattle, hogs, etc.) and another file contains information on "STOCK" (in the sense of "shares", with data pertaining to IBM, General Motors, etc.). If you sit down at the terminal and ask for "STOCK INFORMATION," you will get information you want, plus information you *don't* want:

```
"HOGS       14678 . . .
 IBM        2/3478/87.62/. . .
 SHEEP      12345 . . .
 CATTLE     98362 . . .
 GEN MTRS   4/9231/93.46/. . ."
```

Conversely, if you sit down at the terminal and request information on "CARS" manufactured in the United States, and get the response "REQUESTED DATA NOT FOUND - CARS = 0", it could be because File 1 contains information on "AUTOMOBILES, etc. (Foreign)" and File 2 contains information on "VEHICLES, etc. (U.S. manufactured").

So you can see that a lot of hard thinking goes into getting the terminology right, instead of just dumping all the information into the data base and causing retrieval problems later. Another consideration is that the contents of the data base for the Dictionary/Directory must have amplification information concerning every level that is to be described (see Fig. 1).

DATA LEVELS

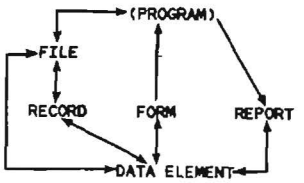


Fig. 1

In order to have reference terms with which we can relate, we shall refer to specific data levels as *entities*. Entities will be distinguished from one another by *attributes* -- descriptors that identify or characterize entities and help to establish relationships within the entity or among entities). The following are some of the attributes:

Identification

- Name
- Abbreviation
- Synonym(s)
- Reference(s)

Description

- Narrative
  - Purpose
  - Scope
- Physical
  - Sequence
  - Size
- Organizations
  - Responsible organization(s)
  - User organization(s)
- Dates
  - Implementation
  - Change
  - Other

(Etc.)

Fig. 2 illustrates some of the entities and their hierarchical relationships. The data

entities include "data base/file," "record," "data element," etc., which describe storage information. All entities above the data levels are the functional or management entities and, you might say, provide the reasons why each data element exists. By that I mean that there is a functional reason -- or should be -- for each data element, and that requirement is dictated by one of the entities above the data levels. Attributes form the actual contents of the Data Dictionary/Directory data base. We therefore would have attributes for every entity used by DDO to accomplish its data processing.

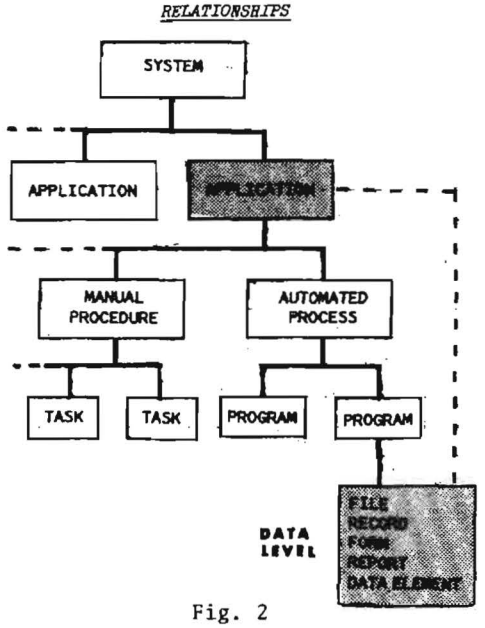


Fig. 2

Fig. 3 is a representation of the dictionary/directory data base -- not how it is actually constructed, but how we visualize it functionally. The

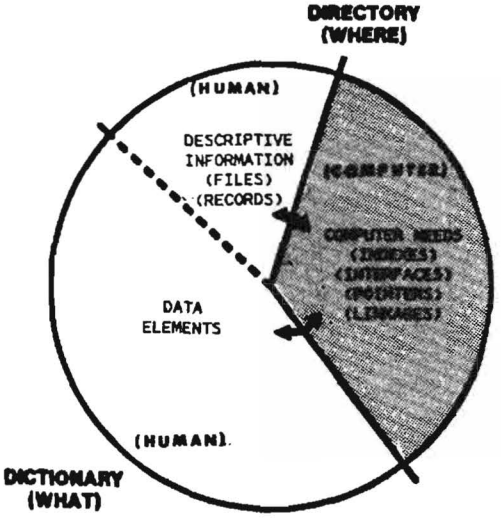


Fig. 3

"what" information is made up of the attributes describing the data elements, and the "where" information" is made up of the attributes describing the records, file, etc. The human aspects I refer to are those descriptions and definitions which are input or used by individuals in a form they can understand. And, finally, the figure shows the machine needs for the computer to maintain the system and the relationships among the entities. The arrows going in both directions indicate that the software uses information *from* the user, as well as providing information *to* him.

Project UTENSIL

I've covered the description of a data dictionary and a few of the uses that DDO could make of such a system. Now I should give you some background on Project UTENSIL. A task force was established by DDO in July 1976 as a result of an A memorandum suggesting the creation of a Data Dictionary/Directory for DDO. The task force forwarded a statement of requirements to C (now T) in February 1977. The project is currently in the fact-gathering and problem-specification stage.

The potential of a DDO Data Dictionary/Directory as a tool is limited only by the degree of commitment to the concept that data is a resource to be managed like people, money, or any other resource. Initially, the DDO system at minimum will be aimed at providing a central computerized resource of information about DDO data. It will provide a capability to show data relationships to all levels contained in the data base, with retrieval capabilities. As with most on-line systems, there will be input and update capabilities to maintain a current file.

The design of the system must be such that new capabilities can be added without affecting those incorporated in the original design. Therefore, I believe we have to make a careful evaluation of the full potential of a data dictionary/directory for DDO to assure that we design the initial system to be flexible enough to allow for any possibility. The fact that we already have several dictionaries available in the Agency proves that dictionaries are feasible, but at the same time it demonstrates that, given the lack of central management, each dictionary has its own merits and shortcomings. What we need is clear, concise direction from management concerning its goals with regard to managing data. Then we can clearly plan not just the short-term goals of a dictionary, but also the data dictionary/directory system as a tool for the long term.

The data dictionary/directory can be just a glossary of terms used by only a few technicians, or, with good planning, it can develop into a tool to be used at all echelons. The uses of a data dictionary/directory system and its data base will be limited only by the ingenuity of its users.

[redacted]

G51

Every time someone defines a "bigamist" as what an Italian calls a dense fog, I am reminded of the article by Robert E. Gould in the December 1975 CRYPTOLOG ("Linguists from the Melting Pot"). The author's main point was that the Agency's dream of recruiting successful translators from ethnic neighborhoods was being frustrated because the aspirants were not working out. Using examples taken from English as spoken by Italian immigrants, he claimed that "anglicisms" had polluted the applicants' *foreign* language so badly that they did not have a chance of succeeding. What he asked those potential translators from ethnic neighborhoods, in effect, was not "What'sa matter? You no spicka da English?", but, rather, "What'sa matter? You no spicka you owna language?"

That idea struck me funny at the time because I could think of no real reason why a person from an ethnic neighborhood should not be able to do well in the Agency. It certainly was not because the applicant would not know the special vocabulary found in Agency work. The Portuguese I had learned in Brazil as a missionary was very different [redacted] and yet I somehow passed the proficiency test. Surely a child growing up in an ethnic neighborhood would learn the *patterns* and many of the idioms of the "foreign" language. With a mastery of those patterns and idioms, the new hire with an exotic surname could be given a glossary of special terms and be expected to do well, no? Apparently not. But why?

A year after reading the article I discovered the reason why the Agency shouldn't be able to find a rich source of good translators in ethnic neighborhoods. For technical reasons I was left for a while without much to translate, and I finally decided to go to the Learning Center for something more to do (besides, the people there are very nice). I found a course (EG 421) entitled "Effective Writing" and started watching television. To my delight the heart of the 6-hour course explained how to edit and unscramble gobbledygook. That was precisely what I needed.

[redacted]

The impact of taking the course was that, although I hadn't been able to do much translating since failing my first attempt at Part II of the PQE, I passed on the second try. EG 421 was the only visible influence that could have made the difference.

So what does EG 421 have in common with hiring linguists? Well, the prerequisite for the course was that anyone wanting the course was supposed to have job duties requiring extensive writing. It hadn't dawned on me that I was doing a lot of writing; after all, my work was to reduce material written by someone else into English. I had not been doing any composition per se, just translating. No wonder I had failed the PQE! I finally passed it when I began to think of myself as a technical writer whose material is usually dictated by, but sometimes only inspired by, the Portuguese it represents. I rarely had trouble understanding Portuguese; my problem was writing English! (The people who edit my translations insist I still have lots of problems.)

My point is that *writing English is the major portion of a translator's job*: the foreign language is secondary. There are lots of good translators in this Agency whose command of their job-related foreign language is far from native, or even that of a college graduate. This means that if an applicant with a childhood foreign-language background doesn't do well, it is not because his neighborhood was polluted with anglicisms; it is because his English was polluted (or in some other way deficient, like mine). In other words, "*That'sa* matter! He no write-a da *English*!"

Having diagnosed the problem, it's now time to prescribe the remedy. If the Agency ever needs to recruit translators again (I specify translator, a person who writes English -- very different from a transcriber, a person who writes some other language -- it was a sad mistake to confuse the two and mislabel them both "linguists"), it should stop restricting the search to language majors. The ideal recruit really might be a Journalism or English major with a Language minor, and some course work in International Economics and Political Science.

My degree is in Economics. Because of that, no one ever asked me if I wanted to go to the D.C. area as a GS-7 and use my Portuguese. No one asked my roommate, who spoke German, to come to NSA either; he was a chemistry major. I knew lots of people who never took college courses in

the foreign languages they spoke fluently (Indonesian, Japanese, or Korean, in addition to French, German, Italian, or Spanish) because they felt they could never make a living with their language and went into other fields; and because they were not language majors, they were not interviewed by NSA. People like this should be talked to by Agency recruiters who are searching for translators. Perhaps recruiters could find people like this with ads in campus papers saying something like "If you got an 'A' in Freshman English and know a foreign language, Uncle Sam wants you!"

An additional thought on recruiting linguists: high ambition in an applicant should be viewed as a criterion for *non*selection. This view contrasts sharply with the one expressed by Daniel G. Buckley in another Agency publication ("Can A Linguist Development Program for High School Graduates Work at NSA?", *Cryptologic Spectrum*, Winter 1977). The problem with highly ambitious or highly motivated people, whether college degree holders or Agency-trained high school graduates, is that they expect (and emotionally require) high productivity to be rewarded, and government service is not set up to give rewards for productivity. The result is that highly motivated linguists begin to look outside the Agency for advancement when they see little opportunity for promotion inside. If the object is to recruit career linguists, no more than a moderate amount of ambition or motivation should be allowed in a recruit.

A word of warning is in order. Due to the accelerated decline in the number of college graduates who can actually write well in English, the Agency may find itself in a position with translators similar to the one it is in now with programmers and engineers: as the general demand for good writers increases in the economy, translators will gain more outward mobility, not for their ability to understand foreign languages, but for their ability to write plain English. Then the Agency may have a "language problem" that will make today's situation pale in comparison.

~~(C)~~

NEWS OF THE C.A.A. (Communications Analysis Association)

By W.E.S.

Busy Week in March

CAA had a busy week in March. On Wednesday, 1 March, [redacted] spoke in the Auditorium, drawing upon her years of experience in the White House Situation Room.

On Thursday morning, 2 March, the CAA Operational Briefing series featured [redacted]

On Thursday afternoon, the CAA Board held its monthly meeting (by the way, we're back to monthly-changing rooms again), and we spent a large part of our time talking with Sergeant [redacted] and Dan Buckley (M09) about how the CAA might assist and encourage professionalization among civilians and military.

Then, on Friday, 3 March, the CAA's Special Interest Group on Cryptologic History had a session on "Oral History," featuring Dave Goodman (former NCS History Fellow) and Art Zobelein. The presentation included an introduction to the principles and techniques of oral history, and a description of the oral history program here at NSA. ~~(C-CCO)~~

P.L. 86-36

Winner in Logo Contest

It was grueling work, but the board knows its oats. After much animated discussion about the wide variety represented in the entries, the CAA Board finally made up its mind. Out of the sixty-plus entries in the CAA Logo Contest, the winning entry came from [redacted] A645, who will receive a book of his choice. (More about the winning entry next month.) (U)

P.L. 86-36

CAA Presentations

Our Program Committee, chaired by [redacted] has been busy too. The committee has lined up the following speakers:

10 May Admiral Inman
21 June [redacted] (subject: Interstellar SIGINT)
(?September?) [redacted] N1

If you have any ideas for other presentations, let Earl know. (U)

P.L. 86-36
EO 1.4.(c)

Meet Linda!

The newest CAA Board Member is [redacted] a COMSEC Analyst in S1. She earned an AB in Psychology at Gettysburg College and came to work here at NSA in 1966.

P.L. 86-36

Her cryptologic experience has included assignments on the staff of the National Cryptologic School, where she was involved with CY-001 and CY-300; in B Group, where she was associated with the [redacted] and in S, where she became a COMSEC Intern.

EO 1.4.(c)
P.L. 86-36

She is certified in Traffic Analysis and COMSEC. (Inside tip: Ask her about collies.) ~~(C-CCO)~~

CAA Board:

President	David Gaddy	3247
President-elect	Frank Porrino	5879
Secretary	[redacted]	8025
Treasurer	Tim Murphy	3791
Board members:	[redacted]	4935
		5991
		3573
		3369

(U)

Four Easy Steps for Joining the CAA:

1. Get a membership card from any of the members of the CAA Board.
2. Fill it out.
3. Attach $1.00 ("Is *that* all it costs?" "Yes, Fred, that's all it costs!")
4. Mail card and money to Tim Murphy, B09.

(U)

P.L. 86-36

The Joys of UNIX

P.L. 86-36

[redacted] P13/R53

For the last 6 months or so, I have been experiencing a new and (in my jaded and cynical mind) totally unexpected turn of events in working with computers. I have actually been enjoying the process of using a computer system: the R53 MYCROFT (PDP 11/70 under the UNIX operating system). After several years of being turned off by discouraging encounters with various computer systems, some involving on-line terminals, I find UNIX, and in particular the RAND editor, restoring much of my lost faith in the value and promise of computers.

I have found myself choosing to do at the terminal more and more things that I would once have done with pencil and paper, with a typewriter, or with cumbersome POGOL procedures on the IBM 370. Even more important, I have chosen to undertake many things that I would hardly have considered before; the convenience and accessibility of UNIX makes new things, or old things done in newer ways, seem pleasant, challenging, and possible rather than prohibitively painful, burdensome, and remote. I have been so struck by the dramatic contrast between my former feelings of disgust and discouragement and my present optimistic and positive feelings about UNIX (and, by extension, about computer technology in general, and the tasks I perform with its aid), that I have spent some time in considering which specific features give UNIX its remarkable value for me as a user.

I felt that some informal comments on this topic -- ways to make a computer system more supportive and hospitable to its day-to-day users -- might be of general interest. Our Agency is becoming more and more committed to on-line, interactive systems. Larger and larger numbers of users will soon be attempting to accomplish an increasingly broad and heterogeneous set of tasks on an ever-growing spider web of intersecting networks. The question of how to design and maintain a "friendly" user interface linking a wide range of users to a wide range of computing facilities is becoming increasingly crucial. For one set of users with varied needs, UNIX appears to have provided one good answer.

First Joy

I would like to set the stage with a bit of history -- a rapid glance back over my own experience with computers as an applications programmer since 1951. I am sure many readers will recognize some of the stages of computer usage at NSA that I mention in passing, and will perhaps also recall similar reactions to them (though many will not, perhaps, have been bothered or pleased by the same developments that bothered or pleased me).

When I first began programming on ATLAS I, I felt that programming was a supremely enjoyable and challenging activity. Debugging was done on the computer, in octal; commands were numbers, as were all addresses in memory, and addresses were fixed. Programmers were also operators, and learned as much as possible about the hardware of the computer, since we had to demonstrate and prove each hardware error to the maintenance men before they would fix it. Input and output were on punched paper tape, printed out on a teletypewriter, corrected with sticky tape and a hand punch. Programmers could understand and get at everything, and we could carry out every step of coding, debugging, and running our programs at our own pace and on our own terms, using simple equipment directly accessible to us. Given a chance to vote on whether we wanted an assembler language for ATLAS I, we voted it down. The "manual," consisting of two mimeographed 8 x 11 inch sheets, was a miracle of clarity and succinctness which I have never since seen equaled: it simply listed exactly what each command did with each bit in each register. In any case, we all knew most of it by heart. Who needed an assembler?

As successive "new" computers came and went, things necessarily got more and more complicated. Assemblers, compilers, and subroutine libraries came along, and operating systems began to assume increasing importance. Addresses became relative or relocatable (so we had to add a base address to everything before we could read our octal dumps), and magnetic tape replaced paper tape for input and output. Programmers were banished from the machine area to the outside of a counter, and a new hierarchy of operators and systems specialists reigned supreme "backstage." Computers rose on the horizon and fell away to make way for still newer ones -- 704, 705, 709, 7090, DCS (to name those which which I myself became most familiar). Still, until the advent of IBM's "third generation" -- the 360s and 370s -- and time-sharing, the changes involved primarily a slow accretion of added features which programmers could assimilate a step at a time. I myself still felt that I understood DCS hardware and software almost as well as I did for the earlier machines, and I still felt that it was worthwhile trying to do so (reading and studying maintenance and software reference manuals to learn as much as possible).

Joy Abating

With the coming of the 360s, there seemed to

be an abrupt discontinuity. Suddenly the "manuals" a programmer needed to study if he really aspired to understand the system had stretched out to fill a rack 10 feet long. The workings of the complex agglomeration of hardware components and peripheral devices in ever-varying configurations were all buried under endless layers of software comprehensible only to "the IBM men" and a very small number of others who chose to specialize full-time in these arcane matters. For programmers whose interest and knowledge were centered around an application rather than programming as such, the effects of this sudden increase in complexity, coupled with a loss of understanding and control, could not fail to be discouraging.

Against this changing background, my own experience was, for the most part, one of growth to accommodate the added complexity. Up until and including DCS (again, in the IBM computer series that I know best), most of the changes seemed to bring improved capabilities without too much of a sacrifice for the user in terms of convenience and control over what was happening. After the 360s arrived, however, my experience began to be one of progressive deterioration in my ability to get what I wanted out of "the system" (the hardware, the software, and, no less important, the "fleshware" -- the people behind the computer installation and their ways of dealing with me as a user).

I became, in fact, less and less of a programmer at all, and more and more simply a procedure-writer who tacked together canned routines or previously debugged POGOL steps to do dull things in a dull way. Since it invariably took me 2 or 3 days just to catch up with all the control-card and rule changes (transgression of which invariably resulted in a canceled run accompanied by little or no helpful information), correct JCL errors, achieve the necessary two valid POGOL listings (a compile and a "go" listing), and cope with all the other things that usually went wrong with FILE cards to keep me from reading my input tape and getting an output tape, it hardly ever seemed worthwhile to try anything the least bit conceptually challenging. I was just glad to get a job done, any way I could!

To sum up this quick sketch of my own view of the trends in NSA computer technology as seen through the eyes of a day-to-day user, I recall an early period of maximum accessibility, complete control and understanding of the computer by the programmer. This was followed by a period in which hardware, software, and the human procedures within which these were embedded became increasingly complex; while many features were removed from the programmer's direct control, the added power and conceptual richness of the facilities at my disposal more than balanced these losses. For me, at least (and, to judge by many comments I have heard, for many others as well), the coming of the 360s upset the balance of power greatly to my disadvantage. There was a lot of computing power available, and POGOL, in particular, was always a convenient, useful tool for accomplishing the data processing functions I needed. Unfortunately, so many people were trying to do so many things with the system (some of them apparently mutually incompatible at times!) that some of us were unable to get much out of it. Thus, for me, the time during which I used the IBM 360 and 370 systems was a very discouraging nadir in my interaction with computers.

Joy Regained

It was at this point that I had the good fortune to discover MYCROFT, UNIX, and the RAND editor. Now, suddenly, I have the best of both worlds -- the illusion of having the computing resources all to myself at the terminal (though many others are enjoying the same experience at the same time), with all the power and richness of a modern computer. Once again I have a chance to understand some of what is going on "under the cover" if I wish to make a reasonable effort to do so; in the office where I work there are helpful and patient people who understand the system and can aid me when the documentation is not enough. I can call on a wide variety of programming languages, and I can also call up generalized functions (sort, select, dedupe, translate or convert, spelling check for English words, and several report generators), all in a very simple and flexible manner. I can create new files and execute my own programs or generalized functions on them directly and easily. The whole system is consistent and unified so that I can quickly learn to use it at my terminal. If I have been able to use and enjoy this system, I am certain that anyone else could do so at least twice as quickly (since I have always had a very hard time learning any new programming language and usually require a long time to feel at all comfortable with it).

The key feature of the MYCROFT system for me, and probably for many other users, is the RAND editor. With the editor, I can write a program, jot down rough notes, or draft a report, placing it in a UNIX file. I can then immediately attempt to compile and execute the program, find the errors, go right back into the editor to make changes, rerun the changed program, re-enter the editor, and so forth, until I have either checked out the program or else decide to leave my terminal (to search for food or water or to satisfy some other basic need). In fact, I have once gain the same ideal debugging situation I enjoyed back in the days of ATLAS I. When I do leave the terminal in the middle of this process, I can rest assured that my files will usually be safe, and will reappear when I log on again just as I left them (a certainty that I never had with the other on-line system I tried). After roughing out a report or some initial jottings (an outline, for example), I can come back to the terminal and rapidly reshape and refine the draft, or fill in the outline as easily as I could with pencil and paper, and

more effectively.

Many readers may be taking issue with me, somewhat as follows: "Aren't these just the things anybody can do with a system like LODESTAR, TSO, or CANDE? What's so special?" I am certain that these systems have many advantages, and would like to see an informal write-up on their good points, seen strictly from the applications-oriented user's point of view. It should be remembered, however, that UNIX operates on a "minicomputer," in contrast to the large-scale systems mentioned above. UNIX provides a remarkable amount of power, coupled with an outstanding user interface, all within a computer systems which costs less than $300,000 -- about as much, I am told, as the disc storage alone of the big IBM, CDC, or Burroughs systems.

I would like to describe for the interested reader some features of UNIX which I find most helpful. First and foremost, the RAND editor is beautifully designed from a human-factors point of view. No other software tool that I have ever used or studied can equal it in this respect. Most other editors are "line editors," requiring that editing be carried out on lines specified by number, as was usual with card-oriented file-update procedures. These line editors also require that some set of commands ("REPLACE," "FIND," "DELETE," "MODIFY") be keyed into the terminal, with the necessary strings to specify the sought string, the replacement string, and so forth. These commands, also, have syntax rules which must be learned and which are easily transgressed. Transgressing the rules brings upon the user the need to rewrite the command (edit the command to the editor!) and try again.

In contrast to this line-oriented, programming-language-like type of editor, the RAND editor allows the user to do most things by pressing a single key. Pressing a special "ARG" key permits the next integer or character string to be fed as a parameter to the function designated by any of the other keys. Thus, a user can accomplish many complex actions by simply pressing "ARG," keying in a number or a string of letters depending on the action desired, and then pressing the single key that stands for the action. This is all the "syntax" that has to be learned, and it is consistent over the whole set of actions provided by the editor. In fact, I can guess at what will happen with a set of key presses I have not tried before, simply by extrapolating from the editor's behavior after the key presses I already know. In many instances of guessing the results of key sequences, I have never so far been disappointed. I cannot think of *any* other programming tool I have seen which can be counted on to behave so transparently, so logically, and so sensibly.

In addition to being designed to behave as a user expects it to behave, the editor even achieves sensible behavior when the user goofs. For example, if I press "ARG," then make a mistake while keying the integer or string parameter, I get a crisp and clear warning and can start over again by simply pressing "ARG" again. Often the user can recover from a mistake by repeating a few simple key strokes. In any case, the mistake and its correction do not spoil or interfere with the text on the screen or the previous correct actions. I remember well how pleased I was when, having written down to the last line of the screen without noticing and then having pressed "CARRIAGE RETURN" to get a new line, I saw the editor obediently roll the window down to display a new page for me to write on, having apparently read my mind.

How It Works

In order to gain a more vivid picture of the editor, let us imagine a user -- me -- sitting down at the terminal to write a report. I have some ideas of what I want to say, but nothing written down yet. I log on (a matter of typing two tiny character strings devoid of "syntax" and in response to two simple prompts). Having decided to call my paper "report," I key in "re report," that is, I call for "re," the RAND editor, to work for me on a file called "report." Since this is a new file which I am about to create, the editor displays a polite message telling me to press a "USE" key to cause the file to be set up. Immediately, the editor then provides me with a "window" labeled with the name "report" and all ready for me to begin writing.

I tab over to a preset margin (which I can change if I wish), and set tabs for indentation or tables if I need them, with a few easy key strokes. Then, since I am in early stages of planning my paper, I begin an outline. I start out:

"1. Introduction
2. Essentials of the problem
3. A summary of past solutions"

At this point, I decide that I want another heading between 2 and 3. I position the cursor anywhere on line 3, and press the "OPEN" key. The editor moves line 3 down one line, and I am all ready to write in the new line 3, then change the old 3 to a 4. After I finish my rough outline, and wish to write in subheadings, I can open up space and squeeze any number of them in with ease. Then, when I am satisfied with the outline, I can write the text in after each heading in the same way. No muss, no fuss, no scratch paper, and the copy I see on my scope is always clean and well formatted, without crossouts or strike-overs.

Suppose I am writing along on a line, and inadvertently continue writing past the right border of the window. A flashing message, and, on some terminals also a beeper, demands my attention, and I see a right-pointing arrowhead warning me of the overflow. I move the cursor back to where I meant to end the line, press

the "DELETE CHARACTER" key, and hold it down while the overflow letters are neatly gobbled up and disappear. If I decide, after I have typed a few words, that I want to leave one out in the middle, I can press the "DELETE CHARACTER" key to squeeze out the unwanted word. To squeeze in some words, I press the "INSERT MODE" key and type them in. Material to the right on the line moves over to accommodate them.

Now, suppose that I wish to incorporate a paragraph from another report, already on a UNIX file in my work space, into this new report. By positioning the cursor along the left margin, pressing "ARG," then keying the name of the old file, and then pressing "CONTROL" and "Z," I can split the screen into two horizontal windows (I can have up to ten windows) and call for the old report to be displayed in the second window while I keep the new one in the first window. I know that the paragraph I want starts with the words" It is obvious that," so I press "ARG," then type in these words, just as I expect them to appear in the text. Then I press "+SCH"; the editor finds the phrase and displays it with its surrounding text in the second window. I count 12 lines in the paragraph; by pressing "ARG," then "12," then "PICK," I copy the entire paragraph into a buffer -- the "pick buffer." I press "CONTROL" and "C" to move the cursor back to my new file in the first window, position the cursor where the paragraph is to be inserted, and press "PUT" -- the paragraph magically appears, and I am ready to go on writing. (This feature is a delight in writing programs: one need only code one version of a routine, then "PICK" it and "PUT" it over and over again wherever a similar routine is desired, changing the details later.) Once I have set up windows and filled them with files, I can switch from file to file in each window, and move the cursor from window to window, with a few quick keystrokes.

Nothing Lost When System Crashes

Now, let us imagine that I have been working at the terminal for about an hour, and suddenly the system crashes. I have not been saving my file as I went along, so I fear that I have lost an hour's work -- work that I might have trouble duplicating from memory. The editor has automatically saved a backup version of my file, "report," as it was before I began the latest editing session, in a file called "report.bak." But that is no help with the changes I have made during the session. With most editors, I would be in a very annoying fix; the RAND editor, however, saves a record of every keystroke I have made during the current session. By keying in a simple sequence of commands, I can call this record in and execute it on the "report.bak" file," so that every motion I made is duplicated until the cursor stops at just the point where it was when the system went down, and I am ready to go again. It is very amusing to watch the cursor scooting around, lines, words, and paragraphs jumping in and out, appearing and disappearing, all untouched by human hands. In fact, the "keystroke file" saved by the editor to produce this re-run of my session is just like any other UNIX file, so that I could get it into a window and edit it with the RAND editor to change sequences of key strokes and thus rewrite the history of my own editing session if I wanted to!

As if this were not enough, there is a "macro" facility in a special form of the RAND editor. This facility permits the user to perform sequences of key strokes and then treat them like little programs; he can designate an entire sequence by one key (for example, "X"), then position the cursor wherever he wants to and press "X" to execute the entire sequence. This would be convenient, for example, in reformatting a fielded file; a "macro" could be defined to accomplish reformatting of a page, then executed for each page.

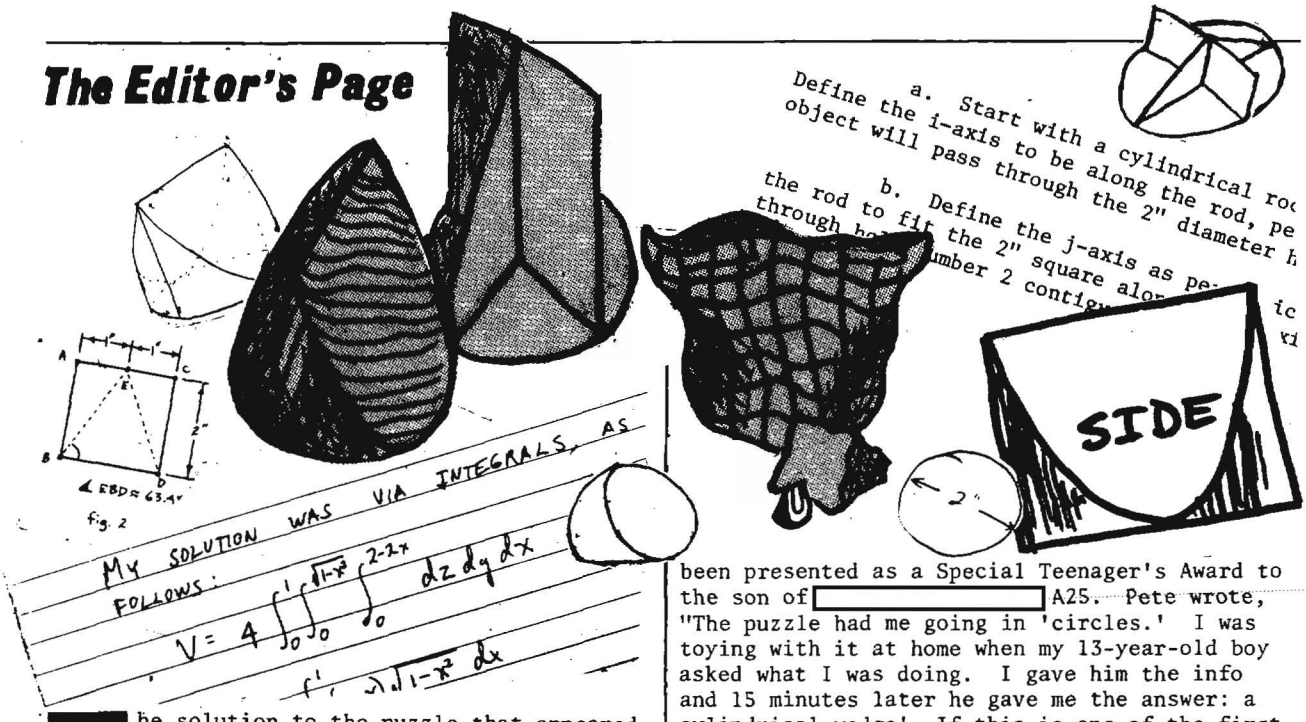
Summary

While the features described above are those that have proven most useful to me, UNIX provides many other advantages for more sophisticated users. For the benefit of readers who may be interested, I will quote a summary of UNIX strengths from a technical report prepared for R53 by a contractor:

> "In general the UNIX world view appears ready-made for user-controllable, multiprocessor systems. The capabilities that make UNIX attractive include process creation (forking), process intercommunication (piping), file directories and referencing via a MULTICS-like tree addressing scheme, the shell concept, command language elegance, the equivalent treatment of system and user procedures, and the user extensible command language."
>
> (Charles Kellogg, "Alternative Architectures for Deductively Augmented Data Management System," TM-6005/000/00 Draft, Systems Development Corporation, December 1977)

I hope that the previous paragraphs have conveyed some of the ease and pleasure of using MYCROFT with UNIX and the RAND editor. I find that I can get much more work done, with far less trouble and effort, using this facility as contrasted to what I could do with other computing systems or with pencil and paper. The RAND editor became familiar to users of the ELROND computer under Programmer's Workbench, and it is available on the KEPLER facility in R17. It is also a part of STEPSTONE II on PLATFORM, and will undoubtedly become a valued tool of many NSA employees. In closing, I would like to emphasize the importance of the "user-friendly" design of the RAND editor, and urge that software designers use it as a model for future systems. With tools like UNIX and the RAND editor, computers can come into their own at last as real aids to human performance.

The Editor's Page



The solution to the puzzle that appeared in the February issue of CRYPTOLOG was printed in the March issue. Because of print-shop deadlines, this is the first opportunity we have to print the names of the winners.

The response to the puzzle was tremendous -- 48 correct answers were mailed in, carried in, or telephoned in (telephone solutions were accepted if the person could explain what he or she had in mind, even without the use of hands). Other than the drawn solutions, representing the easy-to-make version (the three pieces of cardboard), or a cone with a square plane intersecting it vertically, or a piece of a cylinder with two facets shaved off, the editorial office received some "look-alike" descriptions: a vacuum cleaner attachment, a washing machine agitator, etc. We also received some three-dimensional models: an actual piece of a wooden dowel with facets shaved off it (the woodworking equipment to make it must have cost hundreds of dollars); a shaved piece of a pencil eraser; a cut piece of a pencil; a raggedly cut (chewed?) piece of artgum eraser; a piece of graph paper sort of squished into what was an honest attempt at the right answer. And, finally, we got two *offers* of models that never materialized, including an appropriately cut wedge of salami.

The following is an alphabetic list of the first ten persons to provide the correct solution:

[redacted]

David H. Williams, P16.

Flimsy gold-papered cardboard trophies have been sent to the ten winners. Another one has been presented as a Special Teenager's Award to the son of [redacted] A25. Pete wrote, "The puzzle had me going in 'circles.' I was toying with it at home when my 13-year-old boy asked what I was doing. I gave him the info and 15 minutes later he gave me the answer: a cylindrical wedge! If this is one of the first ten, I would like for him to have the trophy." Although the solution was not among the first ten, we are pleased to make this award as a sign of appreciation for spreading the name and fame of CRYPTOLOG.

P.L. 86-36

P.L. 86-36

Other CRYPTOLOG readers who provided the correct solution, but too late to win a trophy (it's a cheap old thing, anyway!), are:

[redacted]

Here it is!

When some of the contest winners showed up in the Editorial Office (that's what I call this mess), I would ask, "Now that you see how easy it is to contribute to CRYPTOLOG, why don't you contribute an article?" Almost invariably, the answer would be, "I might someday, when I get a round TUIT." Okay, then, here's a round TUIT! If anyone -- contest winner or just a casual reader -- is thinking of submitting an article to CRYPTOLOG, cut out the TUIT, fasten it to your article, and send everything to: CRYPTOLOG, P1, Room 2N039. *(Offer void where prohibited)*

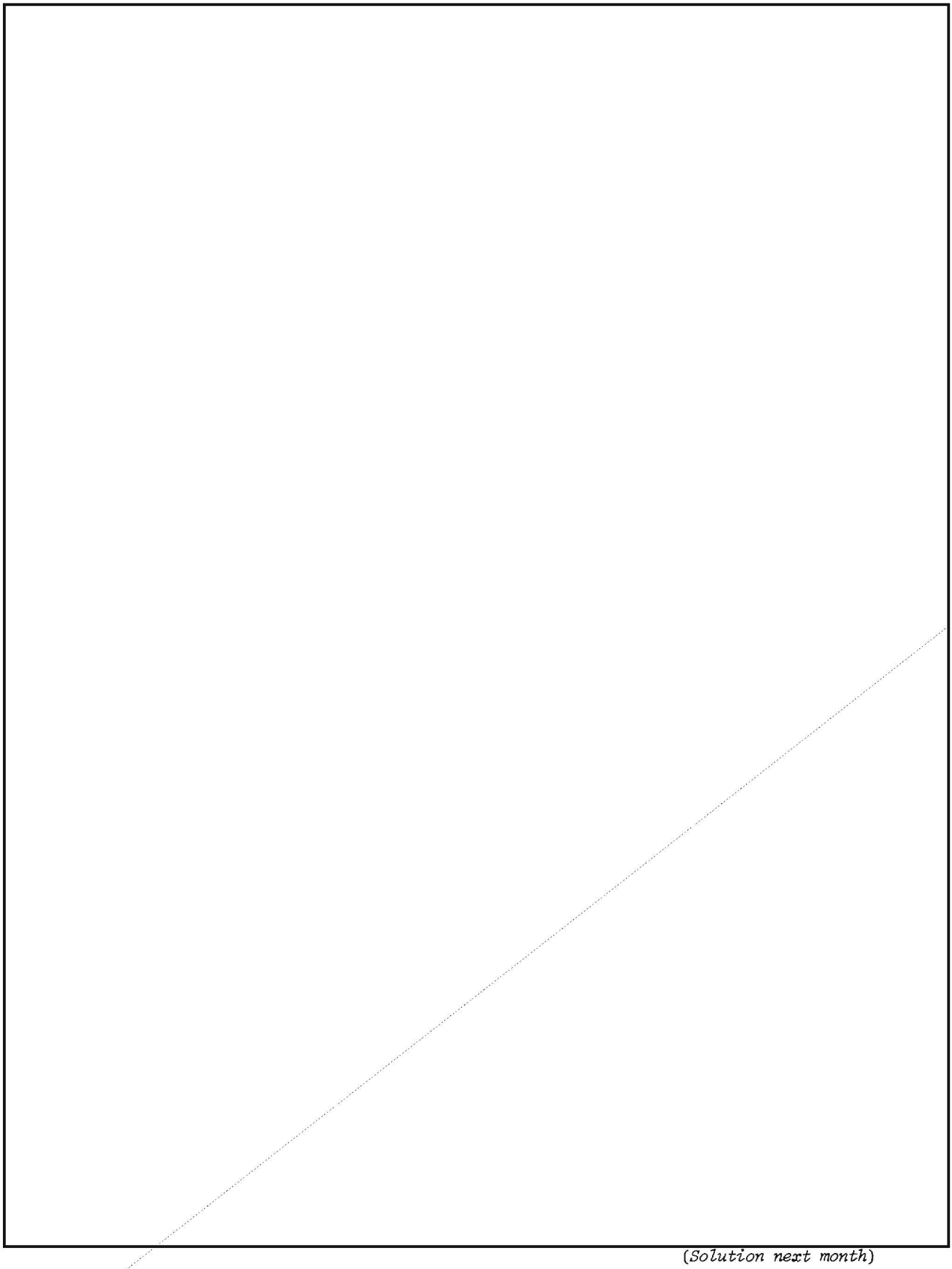
P.L. 86-36

NSA-crostic No. 14

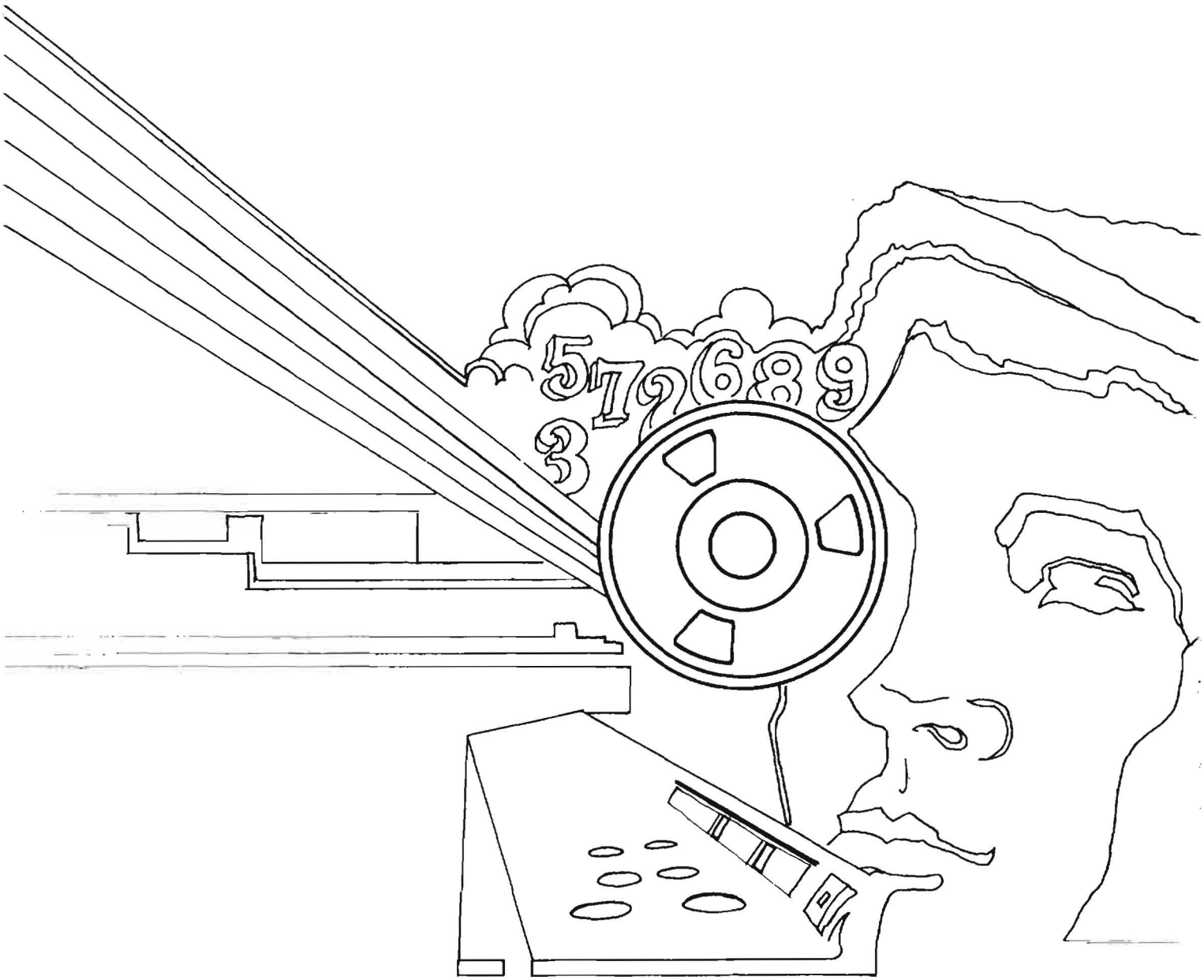
by guest NSA-crostician "Sardonyx"

The quotation on the next page was taken from the published work of an NSA-er. The first letters of the WORDS spell out the author's name and the title of the work.

DEFINITIONS *WORDS*



(Solution next month)


5 7 7 6 8 9
3