

Il Sistema Informativo TP:
la realizzazione della
base di dati.

R. Baraglia - R. Bartoli - O. Signore

Rapporto interno
C83-5

Aprile 1983

Reparto basi di dati e sistemi informativi

CCCCCC	NN	NN	UU	UU	CCCCCC	EEEEEE	
CCCCCC	NNN	NN	UU	UU	CCCCCC	EEEEEE	
C CC	NNNN	NN	UU	UU	CC CC	EE	CNR - Istituto CNUCE
C	NN	NN NN	UU	UU	CC	EEEE	via S. Maria 36
C	NN	NNNN	UU	UU	CC	EEEE	56100 Pisa
C CC	NN	NNN	UU	UU	CC CC	EE	
CCCCCC	NN	NN	UUUUUU	UUUUUU	CCCCCC	EEEEEE	Tel. +39 50 593111
CCCCCC	NN	NN	UUUUU	UUUUU	CCCCCC	EEEEEE	Telex 500371 CNUCE

Indice

Sommario	1
1 - Introduzione.	2
2 - Motivazioni e obiettivi.	3
2.1 - Specifiche informali.	4
2.2 - Descrizione dei dati.	9
3 - Lo schema concettuale.	14
4 - Disegno dell' implementazione.	18
5 - La realizzazione fisica.	21
6 - Conclusioni.	29
Bibliografia	31
Appendice A - Il formalismo grafico DATAID	32
Appendice B.1 - Struttura dati SYSTEM 2000	33
Appendice B.2 - Definizioni DDL SYSTEM 2000	36
Appendice C - Un esempio di sessione	41

Sommario

Viene descritto il lavoro svolto per realizzare un database come supporto fondamentale per l'implementazione del sistema informativo relativo all'area TP di un centro di calcolo erogatore di servizi.

Sono descritte separatamente le varie fasi del lavoro (analisi dei requisiti, progettazione concettuale, progetto dell'implementazione e realizzazione fisica).

Vengono illustrate in dettaglio l'architettura del software realizzato e l'interfaccia con l'utente.

In appendice vengono riportate le definizioni del database e un esempio di sessione.

1 - Introduzione.

La progettazione di basi di dati e' un argomento che ha ricevuto, negli ultimi anni, una notevole attenzione da parte dei ricercatori, e buona parte dell' attivita' svolta e dei risultati raggiunti si possono considerare sistematizzati (si veda, in proposito, l' ottimo [TefR82]).

In particolare, esiste un accordo generale sul fatto che la progettazione di una base di dati passi attraverso quattro fasi:

- analisi dei requisiti;
- progettazione concettuale;
- progettazione logica o disegno dell' implementazione;
- progettazione fisica.

Il presente lavoro descrive, fase per fase, come si e' giunti alla realizzazione di quello che nel seguito verra' chiamato "database TP".

Sullo sviluppo del progetto hanno avuto una certa influenza i risultati dell' obiettivo DATAID del PFI, anche se non e' stata seguita passo per passo la metodologia DATAID-1, divenuta disponibile quando il progetto era in avanzata fase di realizzazione.

Una particolare attenzione e' stata dedicata allo

sviluppo di una interfaccia utente di facile uso, e alla realizzazione di una architettura software modulare e flessibile.

2 - Motivazioni e obiettivi.

Il CNUCE, nell'ambito della razionalizzazione della gestione delle sue risorse hardware e software, ha ritenuto opportuno costituire un sistema informativo per tutta l'area Teleprocessing.

La realizzazione del sistema informativo si proponeva di raggiungere i seguenti obiettivi:

- determinazione dei costi della rete TP;
- pianificazione degli interventi e delle modifiche alla rete;
- identificazione dei guasti piu' frequenti e dei componenti piu' soggetti a guasti;
- evasione sollecita delle richieste degli utenti.

L'opportunità della costituzione di un sistema informativo integrato era dettata anche dal fatto che le informazioni

rilevanti per il raggiungimento di questi obiettivi erano disaggregate, e gestite in modo di fatto indipendente da varie componenti dell' istituto.

2.1 - Specifiche informali.

L'accesso alle risorse di calcolo del CNUCE e' possibile attraverso una rete di collegamenti TP. In fig.1 e' riportata schematicamente la configurazione di tutti i possibili collegamenti.

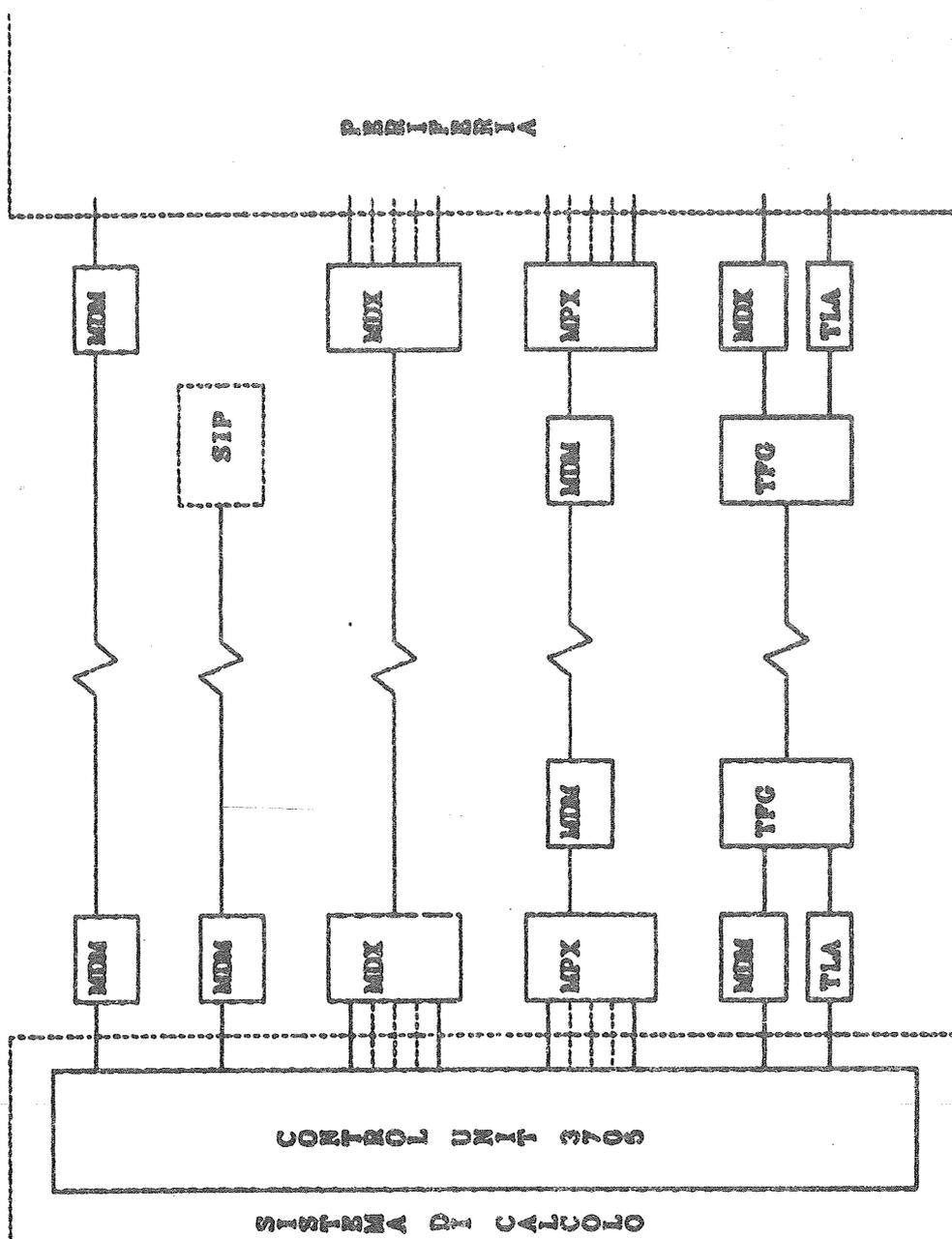


FIG. 1

I simboli presenti in fig.1, quale ad esempio MDM, sono descritti nel paragrafo 2.2.

I collegamenti presentati nella fig.1 sono quelli elementari; si possono avere collegamenti piu' complessi combinando piu' collegamenti elementari. L' elemento periferico puo' essere un terminale oppure l'inizio di un altro collegamento elementare. Per terminale si intende un terminale, batch o interattivo, reale o simulato, oppure un' altra unita' di controllo connessa con un altro sistema.

Il servizio Teleprocessing che il CNUCE mette a disposizione della propria utenza prevede un certo numero di attivita' che possono essere classificate in cinque tipi principali:

- gestione del parco elementi costituenti la rete
- pianificazione degli interventi sulla rete TP
- gestione guasti
- verifica e previsione costi rete TP
- produzione di documenti.

La prima attivita' consiste nel mantenere le informazioni che caratterizzano ciascun elemento della rete, (modem, terminali, linee telefoniche ecc.), utilizzato nei vari collegamenti CNUCE-Utente e viceversa. Tali elementi possono appartenere al CNUCE oppure alla SIP oppure

all'utente collegato. Degli elementi appartenenti al CNUCE viene conservata traccia anche se non sono utilizzati in nessun collegamento. Ciò permette di conoscere la quantità e il tipo di elementi di cui il CNUCE può disporre.

La seconda attività prevede interventi che possono essere di controllo di modifica o di manutenzione della rete TP. Ciò implica la conoscenza di un ampio insieme di informazioni per ogni elemento costituente la rete, quali stato di funzionamento, dislocazione, modalità di utilizzo, disponibilità ed altre.

Ogni tipo di intervento implica la conoscenza degli elementi coinvolti e le modalità di collegamento tra essi.

Un intervento tipico è costituito dalla "problem determination" per individuare l'elemento o gli elementi che determinano il malfunzionamento nel collegamento.

La gestione dei guasti prevede per ogni elemento della rete la registrazione delle informazioni descrittive il guasto ed il tipo di intervento effettuato per la soluzione del guasto stesso.

Inoltre per ogni guasto interessa conoscere le persone e le società coinvolte nella gestione e risoluzione dello stesso.

Dalle informazioni descrittive i guasti deve essere possibile effettuare delle statistiche per la determinazione

dei guasti a cui vanno piu' frequentemente soggetti i vari componenti della rete.

L'attivita' di controllo e pianificazione dei costi TP prevede la verifica del contenuto delle fatture di addebito SIP (emesse a fronte dell' utilizzo degli elementi SIP presenti nei collegamenti) e la realizzazione di previsioni di spesa per utilizzo futuro della rete TP.

La verifica delle fatture SIP consiste nel controllare l'esistenza e le caratteristiche di ogni elemento per il periodo di addebito, cosi' come e' riportato nella fattura. La registrazione degli importi pagati permette di determinare in modo approssimativo quale sara' l'impegno di spesa per l'utilizzo della rete TP in periodi futuri, quali ad esempio il trimestre successivo o l'anno successivo.

Infine, per produzione di documenti si intende la produzione di tutti i consuntivi necessari come supporto per attivita' di tipo manuale, o per archiviazione, o per la compilazione di rapporti da spedire agli organi centrali. Esempi tipici sono l' elenco degli elementi della rete, l' elenco dei collegamenti, i consuntivi di spesa.

2.2 - Descrizione dei dati.

Partendo dalle specifiche informali, attraverso una lunga serie di interviste, di verifiche e di raffinamenti successivi, e' stato possibile identificare i dati di interesse e risolvere tutte le ambiguita'. Contemporaneamente, e' stato possibile procedere ad una classificazione in termini di entita', sottoentita' e attributi. In sostanza, i vari elementi che costituiscono la rete sono stati visti come specializzazione di una entita' identificata come TPE (elemento TP).

Viene qui riportata solo la descrizione dei dati di interesse. Per le specifiche del tipo si rimanda all'appendice B.1 (la corrispondenza tra i dati e i componenti SYSTEM 2000 e' in generale evidente).

<u>TPE</u>	generico elemento TP costituente la rete
	TP-CNUCE
TPEID	costituito dalla concatenazione di una sigla e di un identificatore unico dell'elemento (es. TRM + N. matricola terminale, MDM + n. TD modem, LCU + indirizzo della linea sulla unita' di controllo)
STATO FUNZIONAMENTO	specifica se e' operativo o guasto o in arrivo etc.
UFFICIO COMPETENTE	identifica la sede di competenza per interventi (risoluzione guasti, installazione etc.) sul generico elemento
<u>LINEA</u>	sottoentita' di TPE. E' l'elemento linea

telefonica, identificato dalla sigla LNE
 UNITA' URBANE VERSO IL CNUCE numero di unita' minime fatturabili
 dalla SIP, attualmente una unita' minima
 consta di 200m
 UNITA' URBANE VERSO LA PERIFERIA numero di unita' minime
 fatturabili dalla SIP
 UNITA' EXTRA URBANE scaglione circuito interurbano, da centrale
 SIP a centrale SIP
 QUALITA' indica se e' presente o no l'equalizzatore di
 linea
 TIPO CONNESSIONE indica se lo linea e' di tipo dial o nodial

LINEA C.U.

e' la sottoentita' di TPE che rappresenta la
 linea di connessione CNUCE
 TIPO TRASMISSIONE tipo di colloquio (start-stop, BSC, etc.)
 MODO TRASMISSIONE modo di trasmissione (half duplex o full
 duplex)
 CLOCK descrive se il sincronismo della linea e'
 gestito dall' unita' di controllo oppure dal
 modem
 SPEED velocita' massima di trasmissione
 PROTOCOLLO descrive il protocollo di comunicazione tra
 linea e unita' di controllo
 TIME OUT e', espresso in secondi, il valore del time
 out
 TIPO CONNESSIONE indica se la linea di control unit e' di tipo
 dial o nodial
 SUP.CHANNEL ADDR. indirizzo di sotto canale a cui e' connessa
 la linea. L' associazione e' fissa

TERMINALE

e' la sottoentita' di TPE che descrive l'
 elemento terminale
 CASA COSTRUTTRICE dell' elemento TP terminale
 TIPO indica il tipo del terminale
 MODELLO indica il modello del terminale
 MATRICOLA CASA e' il numero di matricola con cui la casa
 costruttrice identifica il terminale
 CANONE NOLEGGIO UTENTE importo addebitato all'utente per
 l'affitto del terminale
 DATA NOLEGGIO/ACQUISTO data di acquisto/noleggio da parte del CNR
 CANONE AFFITTO/ACQUISTO canone pagato dal CNUCE alla casa
 costruttrice
 MATRICOLA CNR numero di matricola nell' inventario C.N.F.

TERMINALE INTELLIGENTE

sottoentita' di TERMINALE
 NCME LINK nome del remoto connesso da RSCS
 FORMATO SIGNON formato della scheda usata per stabilire il
 collegamento con RSCS
 TIPO LINK indica se e' di tipo NPT, SML, etc.

MODEM

SPEED
MODELLO
CLOCK

sottoentita' di TPE. E' l'elemento modem, identificato dalla sigla MDM
velocita' di trasmissione
indica il modello di modem
indica se il clock e' interno o esterno al modem

MULTIPLEXER

SPEED
MODELLO
CLOCK

sottoentita' di TPE. E' l'elemento multiplexer, identificato dalla sigla MPX
velocita' di uscita del canale verso il modem
indica il modello di multiplexer
indica se il clock e' interno o esterno allo strumento

MULTIPLEXER CHANNEL INPUT
CHANNEL INPUT IDENT.
CHANNEL INPUT SPEED
CHANNEL INPUT TD

sottoentita' di MULTIPLEXER
e' il canale di ingresso al multiplexer
indica la velocita' di trasmissione
e' il numero dell' elemento TP collegato

MULTIPLEXER CHANNEL OUTPUT
CHANNEL OUTPUT IDENT.
CHANNEL OUTPUT SPEED
CHANNEL OUTPUT TD

sottoentita' di MULTIPLEXER
e' il canale di uscita dal multiplexer
indica la velocita' di trasmissione
e' il numero dell' elemento TP collegato

MODEM CANALIZZATO

SPEED
MODELLO
CLOCK

sottoentita' di TPE. E' l'elemento modem canalizzato, identificato dalla sigla MDX
velocita' di uscita del modem
indica il modello di modem canalizzato
indica se il clock e' interno o esterno allo strumento

MODEM CANAL. CHANNEL INPUT
CHANNEL INPUT IDENT.
CHANNEL INPUT SPEED
CHANNEL INPUT TD

sottoentita' di MODEM CANALIZZATO
e' il canale di ingresso al modem
indica la velocita' di trasmissione del canale
e' il numero di identificazione dell' elemento TP collegato

MODEM CANAL. CHANNEL OUTPUT
CHANNEL OUTPUT IDENT.
CHANNEL OUTPUT SPEED
CHANNEL OUTPUT TD

sottoentita' di MODEM CANALIZZATO
e' il canale di uscita dal modem
indica la velocita' di trasmissione del canale
e' il numero di identificazione dell' elemento TP collegato

TFG

SPEED
MODELLO
PORTANTE

sottoentita' di TPE. E' l'elemento TFG, identificato dalla sigla TFG
velocita' di trasmissione
indica il modello di tfg
indica il tipo di portante, alta o bassa

(taglio banda alto o basso)

TFG CHANNEL INPUT sottoentita' di TFG
CHANNEL INPUT IDENT. e' il canale di ingresso al TFG (valori
possibili HI=higt o LW=low)
CHANNEL INPUT SPEED indica la velocita' di trasmissicne
CHANNEL INPUT TD e' il numero di TD dell' elemento TP
collegato

TFG CHANNEL OUTPUT sottoentita' di TFG
CHANNEL OUTPUT IDENT. e' il canale di uscita dal TFG (valori
possibili HI=high o LW=low)
CHANNEL OUTPUT SPEED indica la velocita' di trasmissione
CHANNEL OUTPUT TD e' il numero di TD dell' elemento TP
collegato

PROTOCOLLO protocollo ufficiale del CNUCE

NUMERO numero di registrazione nel protocollo
DATA data di registrazione
OGGETTO oggetto della comunicazione

FATTURA fatture di case costruttrici dei terminali o
bollette SIP ricevute dal CNUCE
NUMERO e' il numero fattura
IMPORTO importo della fattura
MOTIVAZIONE motivazione (es. canone noleggio, intervento
di disinstallazione, riparazione guasto,
canone linea)
PERIODO RIFERIMENTO periodo di riferimento (es. 1 TR 81)

STORIA descrive gli "eventi" di interesse per il
generico elemento TP (es. installazione
presso utente, arrivo, etc.)
DATA INSTALLAZIONE data di installazione presso utente o arrivo
al magazzino CNUCE
CODICE PAGANTE codice di accounting sul quale vengono emesse
le fatture CNUCE per quell' elemento TP
LUOGO DI INSTALLA. nome o sigla dell' istituto presso cui e'
installato l' elemento TP
CITTA' DI INSTALL. citta' in cui e' situato l'istituto in cui
e' installato l'elemento TP
DATA DISINSTALLAZIONE data di fine assegnazione dell' elemento TP
all' utente
MOTIVAZIONE motivo della disinstallazione

GUASTO inconvenienti "aperti" sul generico elemento
TP
DATA data di segnalazione del guasto
ORA ora di segnalazione del guasto

(taglio banda alto o basso)

TFG CHANNEL INPUT sottoentita' di TFG
CHANNEL INPUT IDENT. e' il canale di ingresso al TFG (valori possibili HI=higt o LW=low)
CHANNEL INPUT SPEED indica la velocita' di trasmissione
CHANNEL INPUT TD e' il numero di TD dell' elemento TP collegato

TFG CHANNEL OUTPUT sottoentita' di TFG
CHANNEL OUTPUT IDENT. e' il canale di uscita dal TFG (valori possibili HI=high o LW=low)
CHANNEL OUTPUT SPEED indica la velocita' di trasmissione
CHANNEL OUTPUT TD e' il numero di TD dell' elemento TP collegato

PROTOCOLLO protocollo ufficiale del CNUCE

NUMERO numero di registrazione nel protocollo
DATA data di registrazione
OGGETTO oggetto della comunicazione

FATTURA fatture di case costruttrici dei terminali o bollette SIP ricevute dal CNUCE
NUMERO e' il numero fattura
IMPORTO importo della fattura
MOTIVAZIONE motivazione (es. canone noleggio, intervento di disinstallazione, riparazione guasto, canone linea)
PERIODO RIFERIMENTO periodo di riferimento (es. 1 TR 81)

STORIA descrive gli "eventi" di interesse per il generico elemento TP (es. installazione presso utente, arrivo, etc.)
DATA INSTALLAZIONE data di installazione presso utente o arrivo al magazzino CNUCE
CODICE PAGANTE codice di accounting sul quale vengono emesse le fatture CNUCE per quell' elemento TP
LUOGO DI INSTALLA. nome o sigla dell' istituto presso cui e' installato l' elemento TP
CITTA' DI INSTALL. citta' in cui e' situato l'istituto in cui e' installato l'elemento TP
DATA DISINSTALLAZIONE data di fine assegnazione dell' elemento TP all' utente
MOTIVAZIONE motivo della disinstallazione

GUASTO inconvenienti "aperti" sul generico elemento TP
DATA data di segnalazione del guasto
ORA ora di segnalazione del guasto

DESCRIZIONE	descrizione del guasto
NOTIFICATORE	nome di chi segnala il guasto
APRIGUASTO	nome di chi apre il guasto
PROBLEM DET. GP.	descrizione della problem determination effettuata dall' operatore
DATA CHIUSURA	data di chiusura del guasto
ORA CHIUSURA	ora di chiusura del guasto
IDENTIFICAZIONE	motivo del guasto
SOLUZIONE	descrizione dell' intervento e della societa' intervenuta
TECNICO	nome del tecnico intervenuto

PERIFERIA

	"entita' debole" che rappresenta l' elemento TP immediatamente seguente nella catena di connessione CNUCE ---> Utente
TIPO	indica il tipo di elemento TP collegato
IDENTIFICATORE	identificatore univoco dell' elemento
PORTA	identificatore della porta di uscita, solo per gli elementi TP che prevedono piu' uscite

CENTRO

	"entita' debole" che rappresenta l' elemento TP immediatamente precedente nella catena di connessione CNUCE ---> Utente
TIPO	indica il tipo di elemento TP collegato
IDENTIFICATORE	identificatore univoco dell' elemento
PORTA	identificatore della porta di entrata, solo per gli elementi TP che prevedono piu' entrate

INTERVENTO SPECIALE

	descrive interventi di tipo particolare, di non normale manutenzione, eseguiti sull'elemento TP
DATA RICHIESTA	indica il giorno di richiesta intervento
DESCRIZIONE	descrive il tipo di intervento eseguito
DATA EFFETTUAZIONE	indica quando e' stato eseguito l'intervento

3 - Lo schema concettuale.

Una volta individuati tutti i costituenti della rete come specializzazioni del generico "elemento TP", identificato con il nome TPE, e' stato abbastanza agevole disegnare lo schema concettuale. Infatti, l'identificazione della gerarchia di generalizzazione TPE, per cui ogni elemento della rete (modem, linea, terminale, multiplexer, etc.) puo' essere una e una sola sottoentita' di TPE, trasforma un gran numero di relazioni opzionali in due sole relazioni opzionali: collegamento verso periferia e collegamento verso il centro.

Per meglio comprendere il significato di quanto detto, si pensi che un terminale puo' essere collegato a una linea di control unit, o a un modem, e un modem puo' essere collegato a una linea telefonica o a una porta di multiplexer, che a sua volta puo' essere collegata a una linea telefonica, etc.

Inoltre, per ogni elemento della rete e' necessario o opportuno registrare delle informazioni simili, quali la dislocazione, gli interventi, i guasti segnalati, le fatture relative, etc.

Con l'identificazione della gerarchia di generalizzazione TPE, quindi, le relazioni esistenti tra le varie entita' possono essere schematicamente riassunte nella seguente

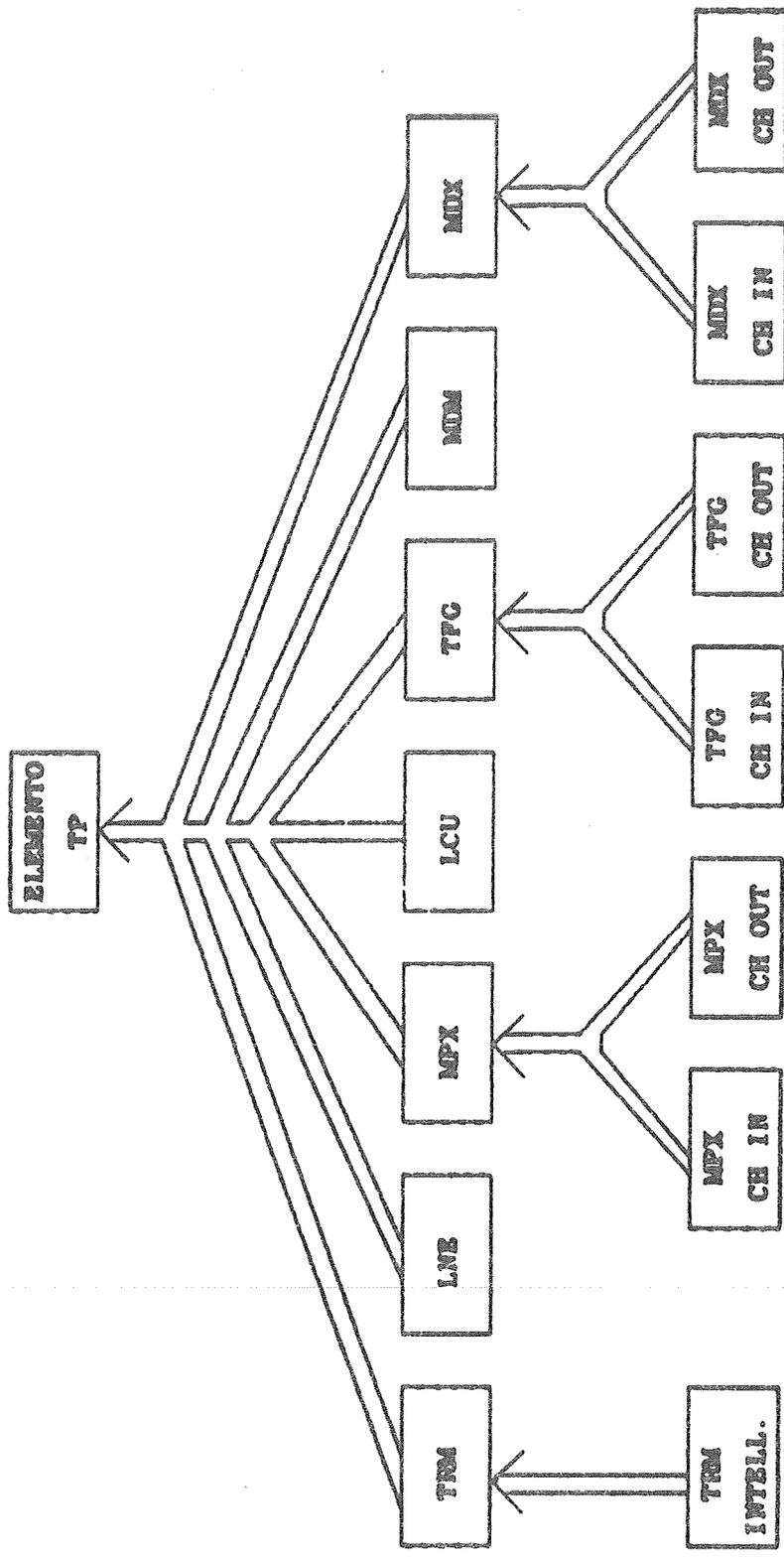
tabella:

===== Dominio =====	===== Codominio =====	===== Nome della relazione =====	===== Cardinalita' =====
TPE	GUASTO	LAMENTA	1:N
TPE	PROTOCOLLO	E' CITATO IN	N:M
TPE	STORIA	STORIA	1:N
TPE	TPE	E' CONNESSO A VALLE	N:M
TPE	TPE	E' CONNESSO A MONTE	N:M
TPE	FATTURA	PAGA	1:N
TPE	INTERVENTO	SUBISCE	1:N

8

La rappresentazione dello schema concettuale secondo il formalismo Entity Relationship Arricchito (vedi appendice A) e' in figura 2a e 2b. Per chiarezza, la figura 2a riporta lo schema concettuale senza le sottoentita' di TPE, la cui gerarchia di generalizzazione e' riportata separatamente in figura 2b.

0



Legenda:

TRM = Terminal
 MDM = Modem
 TPE = Elemento TP
 MDX = Modem

LNE = Linea telefonica
 LCU = Linea control unit
 MPX = Multiplexor
 TFC = TFC

fig. 2b

• ENTITY-RELATIONSHIP •

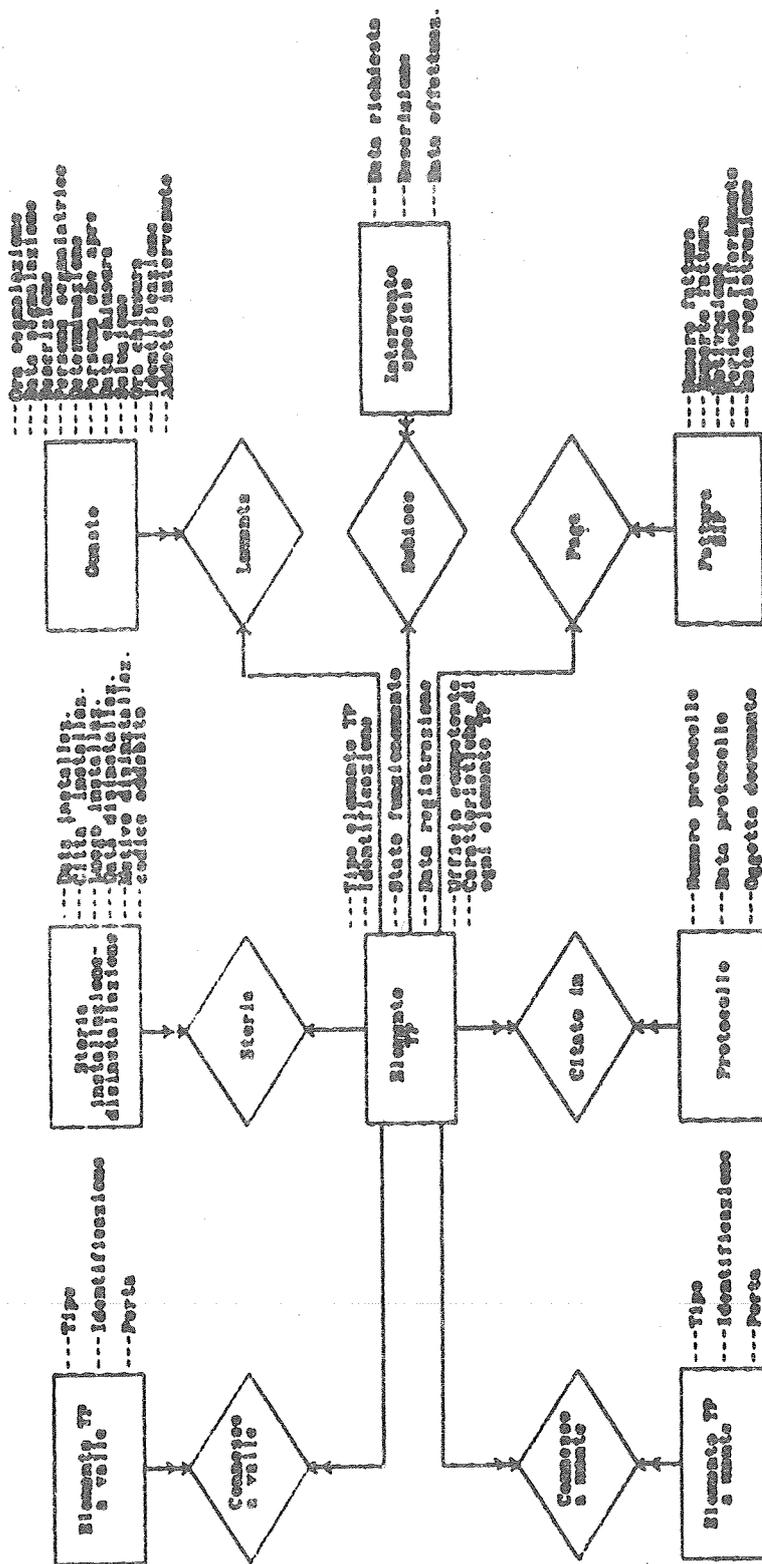


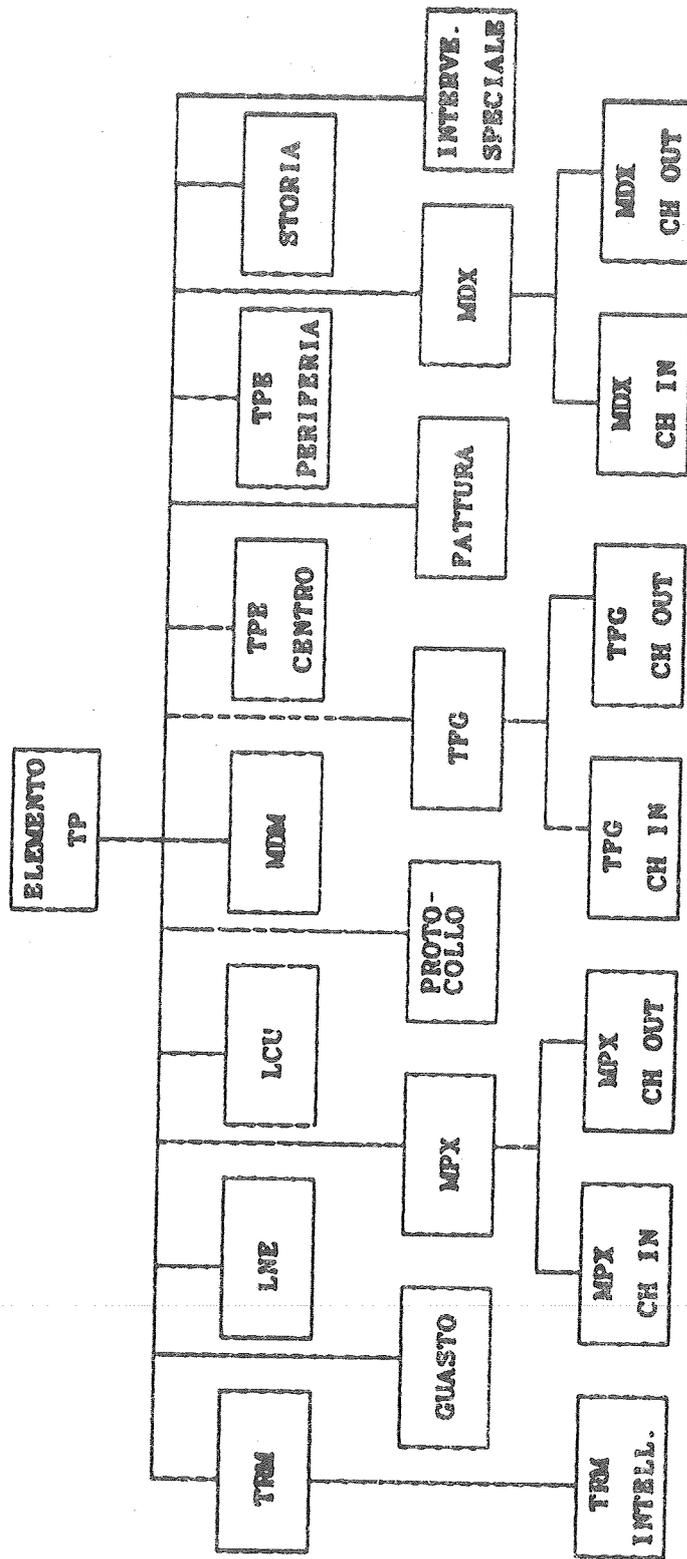
fig. 2a

4 - Disegno dell' implementazione.

Lo scopo della fase di progettazione logica e' quello di trasformare lo schema concettuale in uno schema logico in cui le strutture dei dati siano espresse secondo il modello logico utilizzato dal DBMS usato per la implementazione, in questo caso il SYSTEM 2000.

In questa fase sono stati quindi affrontati i problemi della traduzione dello schema concettuale in uno schema logico di tipo gerarchico e dell' individuazione dell'ambiente software adatto per la realizzazione delle procedure.

In fig. 3 e' rappresentata la struttura logica realizzata.



Legenda:

TRM = Terminale
 MDM = Modem
 TPE = Elemento TP
 MDX = Modem

LNE = Linea telefonica
 LCU = Linea control unit
 MPX = Multiplexor
 TFC = TFC

fig. 3

Per le transazioni di retrieval e' apparso sufficiente l' uso del Self Contained Facility, ossia del query language del SYSTEM 2000. Questa scelta ha permesso di fornire la soluzione adeguata per il raggiungimento di molti degli obiettivi, senza dover ricorrere alla scrittura di software ad hoc.

Va tuttavia precisato che, almeno per il momento, vi sono state alcune resistenze da parte degli utenti a utilizzare il linguaggio, per cui e' stato necessario codificare un gran numero di "stringhe". Questo fatto sembra in accordo con quanto si e' avuto occasione di constatare in altri contesti, in cui e' difficile incontrare un "end user" come viene definito in letteratura, mentre assai piu' frequente e' l'utente parametrico.

Per tutte le transazioni di update, invece, e' apparso subito evidente che l' utilizzo della Self Contained Facility, oltre a presentare dei problemi di accettazione, avrebbe facilmente portato a un deterioramento della qualita' dei dati nel database, dato che si sarebbe avuta una immissione senza il controllo di nessun vincolo (a parte quelli formali di tipo). E' stato quindi necessario sviluppare del software ad hoc in ambiente PLEX (Procedural Language Extension).

Per i dettagli sull' architettura software si rimanda al paragrafo successivo.

5 - La realizzazione fisica.

L'appendice B.1 riporta la definizione della struttura dati realizzata in SYSTEM 2000.

Come si e' detto precedentemente, per la realizzazione delle transazioni di retrieval e per la produzione di documenti si e' fatto esteso ricorso al Self Contained Facility, realizzando un gran numero di stringhe, cioe' di componenti SYSTEM 2000 costituiti da una serie di comandi, utilizzabili da parte dell' utente anche in modo parametrico.

Poiche' la lunghezza di una stringa non puo' superare i 240 caratteri, si e' aggirato il problema ricorrendo alla "nidificazione" di stringhe, per cui in genere una stringa richiama al suo interno altre stringhe, ognuna delle quali, a sua volta, ne puo' richiamare altre.

Per quanto l' identificazione delle funzioni di ogni stringa risulti in tal modo abbastanza complessa, e richieda una buona conoscenza del query language del SYSTEM 2000, va tenuto presente che, dal punto di vista utente, l' utilizzo e' risultato piuttosto agevole, dato che le stringhe richiedono al massimo la specifica di due parametri, e quelle di uso piu' frequente nessuno o al massimo uno.

Per maggiore chiarezza, nella tabella seguente viene riportata una lista dei comandi implementati, con una breve descrizione delle loro funzioni, dei parametri accettati e del significato dei parametri.

In Appendice B.2 viene riportata la lista delle definizioni delle stringhe.

In Appendice C compare un esempio di sessione.

Comando	Descrizione	Parametri	Significato parametri
LISTA par1	Lista degli elementi del tipo par1. Es. *LISTA LCU*	LCU,TFM,LNE, MDM,MPX,MDX, TFG	LCU=linea control unit TRM=terminale LNE=linea telefonica MDM=modem MPX=multiplexer
LISTApar2	Lista dei collegamenti di tipo par2. Es. *LISTACDC*	CDC,CDI,CDU, COM,CTC,FON, ICC,ICU,INT, TEM,CTC	MDx=modem canalizzato CDC=collegamenti diretti su concentratore CDI=collegamenti diretti interurbani CDU=collegamenti diretti urbani
BOLLApar2	Lista delle fatture relative a elementi TP presenti nel collegamento di tipo par2. Es. *BCLLACTC*	ved.comando *LISTA par1*	COM=impianti commutati al CNUCE CTC=collegamenti tramite concentratore FON=impianti fonia ICC=impianti commutati su concentratore ICU=impianti commutati presso utente INT=impianti interni
RETE TP	Lista rete TP del CNUCE		TEM=collegamenti temporanei CTC=collegamenti tramite concentratore
LISTA LINEE	Lista delle linee telefoniche.		
LISTA FATTURE	Lista delle fatture associate ad ogni elemento TP.		
SITUAZIONE TERMINALI	Lista dei terminali con dislocazione.		
*CONNESSIONI DI (<par1>, <par3>)	Lista identificatori degli elementi collegati.	identificatore di TPE	

Il sistema poteva essere sviluppato o in ambiente VS/CICS o in VM/CMS. In ogni caso, il terminale piu' evoluto a disposizione era un terminale tipo IBM 3270. Si e'

preferito l' ambiente CMS, considerata la maggiore dimestichezza con esso da parte degli utenti.

E' stata sviluppata un' interfaccia che, mediante un uso intensivo dei tasti funzionali, facesse interagire l' utente e il sistema mediante "moduli" ossia mediante lo scambio di informazioni logicamente correlate e contenute in uno schermo.

Per ogni entita' o sottoentita' individuata a livello di progettazione concettuale, e' stato costituito un modulo.

Ad ogni TPE e' associata cosi' una serie di moduli.

L' utente puo' quindi avere, con la base di dati, tre tipi fondamentali di interazione: la modifica delle informazioni contenute in un modulo, l' aggiunta di un nuovo modulo contenente informazioni nuove, la cancellazione di un modulo contenente informazioni non piu' necessarie o errate. In questo modo, l' interazione tra l' utente e la base di dati diviene una "navigazione" attraverso i vari moduli associati al singolo elemento TP.

Date queste premesse, e considerato che di fatto ogni entita' o sottoentita', e quindi ogni modulo, e' stata tradotta in un SUB SCHEMA RECORD (da ora in poi indicato per brevitaa' con la sigla SSR) SYSTEM 2000, e' stato facile identificare come funzioni base sul SSR le funzioni di ADD e di EDIT. La funzione di DELETE e' stata vista come una sottofunzione dell' EDIT, considerato che, per cancellare un

record, e' prima necessario essere posizionati su di esso.

Considerato poi che la funzione di ADD puo' essere richiamata dinamicamente mentre si e' in EDIT, appare evidente che ogni applicazione puo' essere vista come una successione di ADD o di EDIT su SSR.

Ovviamente, quest' ultima affermazione e' vera solo se viene mantenuta la qualita' dei dati, cioe' se si suppone di avere automaticamente la verifica dei vincoli ogni volta che si modifica il contenuto della base di dati.

Sulla base di queste considerazioni, e' stato abbastanza facile concepire una architettura software in cui tutte le interazioni con il contenuto della base di dati avvenissero per mezzo di un unico programma, mentre la logica dell' applicazione era gestita da un altro programma. Di conseguenza, abbiamo una architettura a tre livelli.

Il primo livello e' quello a cui si sceglie l'applicazione, tra tutte quelle possibili, e a cui si ritorna con la funzione MENU (il significato di questa e delle altre funzioni verra' spiegato dettagliatamente in seguito).

Al secondo livello abbiamo un programma per ogni scelta possibile al livello precedente. I programmi a questo livello sono quelli che gestiscono la logica della particolare applicazione, preparando una serie di comandi (stack).

Al terzo livello, infine, abbiamo i programmi che realizzano effettivamente l' interazione con il database.

Nel realizzare questa architettura, tuttavia, abbiamo riscontrato una certa rigidita' del DMI del SYSTEM 2000. Infatti, ogni comando PLEX eseguibile e' composto da una parola chiave iniziale (che specifica il tipo di servizio richiesto), un identificatore di database o nome di subschema record, e una o piu' altre specifiche o variabili di programma.

Non essendo possibile specificare in modo parametrico il nome del subschema record o la clausola WHERE, si ha una proliferazione incredibile di linee di codice per realizzare, spesso, le stesse funzioni su SSR differenti, con i conseguenti problemi di manutenzione del software.

La soluzione al problema e' stata identificata nella scrittura di una routine generale (che contiene tutto il software dipendente dalla gestione del video e tutti i comandi PLEX) che viene "personalizzata" al singolo SSR per mezzo di opportune variabili di PREPROCESSOR PL/I.

In tal modo, la scrittura della routine che gestisce un particolare SSR (e quindi il corrispondente modulo) si riduce a una serie di istruzioni del tipo:

```
%INCLUDE <xxxSSR>
```

in cui i moduli <xxxSSR> sono quelli che realizzano la corrispondenza tra i campi del modulo e i campi del SSR

(eventualmente impostando dei valori automatici o effettuando la verifica formale) e il modulo di verifica dei vincoli. Il modulo di verifica dei vincoli e' l' unico che puo' eventualmente richiedere da parte del programmatore la conoscenza del DML del SYSTEM 2000.

Nella routine generalizzata vengono gestite tutte le funzioni di interazione con la base di dati per mezzo dei tasti funzionali. La corrispondenza tra tasto e funzione e' sempre evidenziata chiaramente all' utente, e l' associazione tra tasto e funzione e' stata mantenuta il piu' omogenea possibile.

Le funzioni definite sono:

EXIT	termina la sessione
MENU	torna all' inizio: scelta delle applicazioni disponibili (cice' passa dal livello 2 al livello 1)
QUIT	annulla tutte le modifiche apportate al modulo in esame (ossia allo schermo corrente)
SAVE	registra nel database il modulo, con tutte le variazioni eventualmente apportate
DELETE	cancella dal database tutte le informazioni contenute nel modulo in esame (cioe' cancella il SSR). Dopo aver confermato l' effettuazione dell' operazione, passa a visualizzare il successivo

modulo dello stesso tipo, se chiaramente e'
previsto piu' di un modulo dello stesso tipo.

FORM durante la funzione di EDIT, ottiene un modulo
vuoto, per permettere l' inserimento di nuovi
dati

NEXT durante la funzione di EDIT, permette la
visualizzazione del modulo successivo (sullo
schermo compare l' indicazione "n DI m")

PREVIOUS e' l' opposto della funzione NEXT

LAST visualizza l' ultimo schema del tipo di quello
corrente

END FORM nel caso in cui lo stack di comandi preveda l'
inserimento di piu' moduli dello stesso tipo,
termina il loop di richiesta di dati per moduli
del tipo corrente

END ENTRY ignora tutti i comandi ancora da eseguire

HELP da' una breve descrizione delle varie funzioni

6 - Conclusioni.

Nel corso dello sviluppo dell' applicazione e' apparsa evidente l' importanza di specifiche complete e chiare. In effetti, si e' giunti alla definizione di tutti gli attributi solo dopo molti raffinamenti. Probabilmente, una maggiore circolazione di documentazione scritta porterebbe a una piu' rapida comprensione e formalizzazione del problema.

La rappresentazione dello schema concettuale con il formalismo Entity-Relationship e' apparso di facile comprensione da parte degli utenti.

L' architettura software realizzata si e' dimostrata soddisfacente, sia per quanto riguarda la mantenibilita' e flessibilita', sia per quanto riguarda l' interfaccia utente.

In particolare, l' accettazione da parte dell' utente finale e' stata buona. Probabilmente questo e' legato all' impiego dei tasti funzionali, alla possibilita' di ripristinare il contenuto di un modulo (funzione QUIT) e alla possibilita' di richiamare in qualunque istante la funzione HELP.

Nel corso della realizzazione, pero' e' stato necessario in qualche modo ovviare alla intrinseca rigidita' del PLEX. La soluzione adottata, mentre permette da un lato di sviluppare con relativa semplicita' ulteriori applicazioni, porta

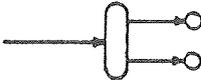
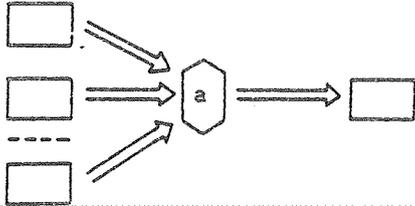
tuttavia a una proliferazione di moduli (uno per ogni modulo o SSR) e quindi impone l' utilizzo di molta memoria centrale.

Ove si procedesse alla stesura di una nuova versione del software, si potrebbe pensare ad alcuni sviluppi. In particolare, la creazione di mappe parametriche, il link dinamico dei moduli software necessari di volta in volta, e una funzione di HELP piu' ricca.

Bibliografia

- [TeFr82] Teory t., Fry j., Design of Database Structures
(Prentice Hall, 1982)
- [CERI83] S. Ceri, editor: Methodology and tools for
database design
North Holland
- [DeAD83] V. De Antonellis, B. Demo: Requirements Collection
and Analysis - in [CERI83]
- [BuCe83] U. Bussolati, S. Ceri, V. De Antonellis, B. Zonta:
Views conceptual design, in [CERI83]
- [CHEN76] Chen P.P., The entity relationship model: toward
an unified view of data, ACM TODS, 1(1) (1976)
9-36.
- [DATAID] Metodologia per la progettazione manuale di basi
di dati - CNR - PFI - Sottoprogetto P2, Obiettivo
DATAID
- [LSM-COB] SYSTEM 2000-The Language Specification Manual For
The COBOL Programming Language Extension (PLEX)
For IBM OS/VS.-1979 MRI System Corporation.
- [LSM-SCF] SYSTEM 2000-SYNTAX GUIDE-Self Contained Facility
For IBM OS/VS.-1979 MRI System Corporation.

Appendice A - Il formalismo grafico DATAID

CONCEPT	REPRESENTATION
ENTITY	
RELATIONSHIP (total/partial)	
RELATIONSHIP (1:1/1:n/n:m)	
ATTRIBUTE (total/partial)	
IDENTIFIER (internal/external)	
REPEATING ATTRIBUTE	
AGGREGATE	
GENERALIZATION HIERARCHY with underlying attribute a	
SUBSET HIERARCHY	
COMPLETE HIERARCHY	

(D.a [Buce 83])

Appendice B.1 - Struttura dati SYSTEM 2000

DATA BASE NAME IS TP

- 1* TPF TYPE (CHAR XXX)
- 2* TPE ID (CHAR X(14))
- 3* TPE STATO FUNZIONAMENTO (CHAR X(11))
- 4* TPE DATA REGISTRAZIONE NEL DE (DATE)
- 5* TPE UFFICIO COMPETENTE (NON-KEY CHAR X(20))
- 6* TPE FREE TEXT (NON-KEY CHAR X(40))

100* TRM (RECORD)

- 101* TRM CASA COSTRUTTRICE (CHAR X(8) IN 100)
- 102* TRM TIPO (CHAR X(6) IN 100)
- 103* TRM MODELLO (CHAR X(10) IN 100)
- 104* TRM CANONE NOLEGGIO PER UTENTE (INTEGER NUMBER 9(7) IN 100)
- 105* TRM MATRICOLA CNP (NON-KEY CHAR X(6) IN 100)
- 106* TRM DATA ACQUISTO/NOLEGGIO (DATE IN 100)
- 107* TRM PREZZO ACQUISTO/NOLEGGIO (INTEGER NUMBER 9(8) IN 100)
- 150* TRM INTELLIGENTE (RECORD IN 100)
 - 151* TRM NOME LINK (CHAR X(8) IN 150)
 - 152* TRM FORMATO SIGNON (NON-KEY CHAR X(20) IN 150)
 - 153* TRM TIPO LINK (NON-KEY CHAR X(20) IN 150)

200* LNE (RECORD)

- 201* LNE N.UNITA' URBANE VERSO CNUCE (NON-KEY INTEGER NUMBER 999 IN 200)
- 202* LNE N.UNITA' URBANE VERSO PERIFERIA (NON-KEY INTEGER NUMBER 999 IN 200)
- 203* LNE N.UNITA' EXTRA URB. (NON-KEY INTEGER NUMBER 999 IN 200)
- 204* LNE FASCIA (NON-KEY CHAR XX IN 200)
- 205* LNE QUALITA' (NON-KEY CHAR X(10) IN 200)
- 206* LNE TIPO CONNESSIONE (CHAR X(6) IN 200)

300* LCU (RECORD)

- 301* LCU TIPO TRASMISSIONE (CHAR XXXX IN 300)
- 302* LCU MODO TRASMISSIONE (NON-KEY CHAR XX IN 300)
- 303* LCU CLOCK (NON-KEY CHAR XXX IN 300)
- 304* LCU SPEED (INTEGER NUMBER 9999 IN 300)
- 305* LCU PROTOCOLLO (NON-KEY CHAR XXXX IN 300)
- 306* LCU TIME OUT (INTEGER NUMBER 999 IN 300)
- 307* LCU TIPO CONNESSIONE (NON-KEY CHAR X(6) IN 300)
- 308* LCU SUB CHAN ADDR (NON-KEY CHAR XXX IN 300)

400* MDM (RECORD)

- 401* MDM SPEED (INTEGER NUMBER 9999 IN 400)
- 402* MDM MODELLO (NON-KEY CHAR X(10) IN 400)
- 403* MDM CLOCK (CHAR XXX IN 400)

500* MPX (RECORD)

- 501* MPX SPEED (INTEGER NUMBER 9999 IN 500)
- 502* MPX MODELLO (NON-KEY CHAR X(10) IN 500)
- 503* MPX CLOCK (CHAR XXX IN 500)
- 550* MPX CHN IN (RECORD IN 500)
 - 551* MPX CHN IN ID (CHAR XX IN 550)

552* MPX CHN IN SPEED (NON-KEY INTEGER NUMBER 9999 IN 550)
 553* MPX CHN IN FREE TEXT (NON-KEY CHAR X(90) IN 550)
 554* MPX CHN IN JTD (CHAR X(14) IN 550)
 560* MPX CHN OUT (RECORD IN 500)
 561* MPX CHN OUT ID (CHAR XX IN 560)
 562* MPX CHN OUT SPEED (NON-KEY INTEGER NUMBER 9999 IN 560)
 563* MPX CHN OUT FREE TEXT (NON-KEY CHAR X(90) IN 560)
 564* MPX CHN OUT JTD (CHAR X(14) IN 560)
 600* MDX (RECORD)
 601* MDX SPEED (INTEGER NUMBER 9999 IN 600)
 602* MDX MODELLO (NON-KEY CHAR X(10) IN 600)
 603* MDX CLOCK (CHAR XXX IN 600)
 650* MDX CHN IN (RECORD IN 600)
 651* MDX CHN IN ID (CHAR XX IN 650)
 652* MDX CHN IN SPEED (NON-KEY INTEGER NUMBER 9999 IN 650)
 653* MDX CHN IN FREE TEXT (NON-KEY CHAR X(90) IN 650)
 654* MDX CHN IN JTD (CHAR X(14) IN 650)
 660* MDX CHN OUT (RECORD IN 600)
 661* MDX CHN OUT ID (CHAR XX IN 660)
 662* MDX CHN OUT SPEED (NON-KEY INTEGER NUMBER 9999 IN 660)
 663* MDX CHN OUT FREE TEXT (NON-KEY CHAR X(90) IN 660)
 664* MDX CHN OUT JTD (CHAR X(14) IN 660)
 700* TFG (RECORD)
 701* TFG SPEED (INTEGER NUMBER 9999 IN 700)
 702* TFG MODELLO (NON-KEY CHAR X(10) IN 700)
 703* TFG PORTANTE (CHAR X(5) IN 700)
 750* TFG CHN IN (RECORD IN 700)
 751* TFG CHN IN ID (CHAR XX IN 750)
 752* TFG CHN IN SPEED (NON-KEY INTEGER NUMBER 9999 IN 750)
 753* TFG CHN IN FREE TEXT (NON-KEY CHAR X(90) IN 750)
 754* TFG CHN IN JTD (CHAR X(14) IN 750)
 760* TFG CHN OUT (RECORD IN 700)
 761* TFG CHN OUT ID (CHAR XX IN 760)
 762* TFG CHN OUT SPEED (NON-KEY INTEGER NUMBER 9999 IN 760)
 763* TFG CHN OUT FREE TEXT (NON-KEY CHAR X(90) IN 760)
 764* TFG CHN OUT JTD (CHAR X(14) IN 760)
 1000* TPECENTRO (RECORD)
 1001* TPECENTRO TYPE (CHAR XXX IN 1000)
 1002* TPECENTRO ID (CHAR X(14) IN 1000)
 1003* TPECENTRO PORTA (CHAR XX IN 1000)
 1100* TPEPERIFERIA (RECORD)
 1101* TPEPERIFERIA TYPE (CHAR XXX IN 1100)
 1102* TPEPERIFERIA ID (CHAR X(14) IN 1100)
 1103* TPEPERIFERIA PORTA (CHAR XX IN 1100)
 1200* GUASTO (RECORD)
 1201* GUASTO ORA SEGNALAZIONE (NON-KEY CHAR X(8) IN 1200)
 1202* GUASTO DATA SEGNALAZIONE (NON-KEY DATE IN 1200)
 1203* GUASTO DESCRIZIONE (NON-KEY CHAR X(100) IN 1200)
 1204* GUASTO PERSONA CHE SEGNALE (NON-KEY CHAR X(20) IN 1200)
 1205* GUASTO PROBLEM DETERMINATION OP. (NON-KEY CHAR X(100) IN 1200)

0)
 1206* GUASTO PERSONA CHE APRE (NON-KEY CHAR X(20) IN 1200)
 1207* GUASTO DATA CHIUSURA (NON-KEY DATE IN 1200)
 1208* GUASTO SOLUZIONE (NON-KEY CHAR X(100) IN 1200)
 1209* GUASTO ORA CHIUSURA (NON-KEY CHAR X(8) IN 1200)
 1210* GUASTO IDENTIFICAZIONE (NON-KEY CHAR X(100) IN 1200)
 1211* GUASTO ADDETTO INTERVENUTO (NON-KEY CHAR X(20) IN 1200)
 1300* BOLLA (RECORD)
 1301* BOLLA NUMERO (NON-KEY CHAR X(10) IN 1300)
 1302* BOLLA IMPORTO (NON-KEY INTEGER NUMBER 9(8) IN 1300)
 1303* BOLLA MOTIVAZIONE (NON-KEY CHAR X(6) IN 1300)
 1304* BOLLA PERIODO RIFERIMENTO (NON-KEY CHAR X(8) IN 1300)
 1305* BOLLA DATA REGISTRAZIONE (NON-KEY DATE IN 1300)
 1400* STORIA (RECORD)
 1401* STORIA DATA INSTALLAZIONE (DATE IN 1400)
 1402* STORIA CODICE ACCT ATTRIBUITO (CHAR XXXX IN 1400)
 1403* STORIA LUOGO INSTALLAZIONE (TEXT X(60) IN 1400)
 1404* STORIA DATA DISINSTALLAZIONE (NON-KEY DATE IN 1400)
 1405* STORIA MOTIVO DISINSTALLAZIONE (NON-KEY CHAR X(100) IN 1400)
 1406* STORIA CITTA INSTALLAZIONE (CHAR X(20) IN 1400)
 1500* PROTOCOLLO (RECORD)
 1501* PROTOCOLLO NUMERO (INTEGER NUMBER 9(5) IN 1500)
 1502* PROTOCOLLO DATA (DATE IN 1500)
 1503* PROTOCOLLO OGGETTO (NON-KEY CHAR X(100) IN 1500)
 1600* INTERVENTO SPECIALE (RECORD)
 1601* INTERVENTO DATA RICHIESTA (DATE IN 1600)
 1602* INTERVENTO DESCRIZIONE (NON-KEY CHAR X(40) IN 1600)
 1603* INTERVENTO DATA EFFETTUAZIONE (DATE IN 1600)

Appendice B.2 - Definizioni DDL SYSTEM 2000

2100* L (STRING (LI C2,C1301,C1302 WH C1 EQ LNE AND C2 EQ *1* AND C1300 HAS C1301 EQ 1400000000 AND C1300 HAS C1302 EQ *2*::))

2101* NF (STRING (CH C1301 EQ *1* * WH SAME::))

2000* CARICA (STRING (*LDLCU*:*LDMPX*:*LDMDX*:*LDLNE*:*LDTFG*:*LDTRM*:*LDMDM*::))

2001* LDLCU (STRING (DATA FILE IS LCU:LOAD::))

2002* LDMPX (STRING (DATA FILE IS MPX:LOAD::))

2003* LDMDX (STRING (DATA FILE IS MDX:LOAD::))

2004* LDLNE (STRING (DATA FILE IS LNE:LOAD::))

2005* LDTFG (STRING (DATA FILE IS TFG:LOAD::))

2006* LDTRM (STRING (DATA FILE IS TRM:LOAD::))

2007* LDMDM (STRING (DATA FILE IS MDM:LOAD::))

2500* LISTA ALL (STRING (*LISTA TRM*:*LISTA LCU*:*LISTA MDM*:*LISTA MPX*:*LISTA MDX*:*LISTA TFG*::))

3000* INIZIA (STRING (DATE FORMAT IS DD/MM/YY:ECHG OFF::))

3001* LCUTITLE (STRING ((80) D(10) LINEE DI CONTROL UNIT))

3003* LCUHEAD (STRING (L(6) TIPO E++TRASMI, B(0), L(6) MODO++SSIONE, P(0), L(6)+CLOCK, B(0), L(6)+SPEED, B(3), L(6)+PROT., B(1), L(9)+TIMEOUT, B(1), L(6)+TIPO+CONN.))

3004* LCUDATA (STRING (C301,C302,C303,C304,C305,C306,C307))

3005* PERIFHEAD (STRING (L(4), B(0), L(15) E' CONNESSO+VERSO PERIFERIA+CN, L(5)++PORTA))

3006* PERIFDATA (STRING (C1101,C1102,C1103))

3007* TPEHEAD (STRING (L(4), B(0), L(15) ID))

3008* TPEID (STRING (C1,C2))

3009* TRMHEAD (STRING (L(13) CASA+COSTRUTTRICE, B(0), R(6) TIPO, B(1), L(10) MODELLO, B(0), L(8) CANONE+NCLEGGIO+UTENTE, B(1), L(1) MATRICOLA+CNR, B(1), L(10) DATA+ACQUISTO+NOLEGGIO, L(12) PREZZO DI+ACQUISTO+O NOLEGGIO))

3010* TRMDATA (STRING (C101,C102,C103,C104,C105,C106,C107))

3011* LNEHEAD (STRING (N. UNITA'+URBANE+LATO CNUCE, B(1), N. UNITA'+URBANE+PERIFERIA, B(1), N. UNITA'+EXTRAURBANE, B(1), FASCIA, B(1), QUALITA'))

3012* LNEADATA (STRING (C201,C202,C203,C204,C205))

3013* MDMHEAD (STRING (SPEED+MDM, B(1), L(15) MODELLO+MDM, B(1), CLCK+MDM))

3014* MDMDATA (STRING (C401,C402,C403))

3015* MPXHEAD (STRING (SPEED+MPX, B(1), MODELLO+MPX, B(1), CLOCK+MPX))

3016* MPXDATA (STRING (C501,C502,C503))

3017* MPXCHINHEAD (STRING (MPX+CH IN +ID, B(1), MPX+CH IN +SPEED, B(1), MPX+CH IN +INFORMAZIONI, B(1), L(11) MPX +CH IN +JTD))

3018* MPXCHINDATA (STRING (C551,C552,C553,C554))

3019* MPXCHOUHEAD (STRING (MPX+CH OUT+ID, B(1), MPX+CH OUT+SPEED, P(1), MPX+CH OUT+INFORMAZIONI, B(1), L(11) MPX +CH OUT+JTD))

3020* MPXCHOUADATA (STRING (C561,C562,C563,C564))

3021* MDXHEAD (STRING (SPEED+MDX, P(1), L(10) MODELLO+MDX, B(1), CLCK+MDX))

3022* MDXDATA (STRING (C601,C602,C603))
3023* MDXCHINHEAD (STRING (MDX+CH IN +ID,B(1),MDX+CH IN +SPEED,B(1),MDX+CH IN +INFORMAZIONI,B(1),L(11)MDX +CH IN +]TD))
3024* MDXCHINDATA (STRING (C651,C652,C653,C654))
3025* MDXCHOUHEAD (STRING (MDX+CH OUT+ID,B(1),MDX+CH OUT+SPEED,B(1),MDX+CH OUT+INFORMAZIONI,B(1),L(11)MDX +CH OUT+]TD))
3026* MDXCHOUHEAD (STRING (C661,C662,C663,C664))
3027* TFGHEAD (STRING (SPEED+TFG,B(1),MODELLO+TFG,B(1),PORTANTE+TFG))
3028* TFGDATA (STRING (C701,C702,C703))
3029* TFGCHINHEAD (STRING (TFG+CH IN +ID,B(1),TFG+CH IN +SPEED,B(1),TFG+CH IN +INFORMAZIONI,B(1),L(11)TFG +CH IN +]TD))
3030* TFGCHINDATA (STRING (C751,C752,C753,C754))
3031* TFGCHOUHEAD (STRING (TFG+CH OUT+ID,B(1),TFG+CH OUT+SPEED,B(1),TFG+CH OUT+INFORMAZIONI,B(1),L(11)TFG +CH OUT+]TD))
3032* TFGCHOUHEAD (STRING (C761,C762,C763,C764))
3033* CENTROHEAD (STRING (L(4),B(0),L(15)E' CONNESSO+VERSO CNUCE+CCN,L(5)++PORTA))
3034* CENTRODATA (STRING (C1001,C1002,C1003))
3035* GUASTOHEAD (STRING (GUASTO))
3036* GUASTODATA (STRING (C1201,C1202,C1203,C1204,C1205,C1206,C1207,C1208,C1209,C1211))
3037* BOLLAHEAD (STRING (N. FATTURA ,IMPOR TO ,B(1),MOTIVAZIONE,B(1),PERICDO,B(1),REGISTRATA+ IL))
3038* BOLLADATA (STRING (C1301,C1302,C1303,C1304,C1305))
3039* STORIAHEAD (STRING (DATA+INSTALLAZIONE,B(1),SU+CODICE,INSTALLATO+ A,B(1),DISINSTALLATO+ IL,B(1),L(20) PER))
3040* STORIADATA (STRING (C1401,C1402,C1403,C1404,C1405))
3041* PROTOCOLLOHEAD (STRING (NUMERO+PROT.,E(1),IN DATA ,B(1),L(20) GGGETTO))
3042* PROTOCOLLODATA (STRING (C1501,C1502,C1503))
3043* LINEEHEAD (STRING (L(15)NUMERO DI TD,B(0),L(60)DISLOCAZIONE DELLA LINEA,B(0),INSTALLATA+ IL,B(1),DISINSTALLATA+ IL))
3501* TRMTITLE (STRING (D(10) LISTA TERMINALI))
3502* LNETITLE (STRING (D(10) LISTA LINEE TELEFONICHE))
3503* MDMTITLE (STRING (D(10) LISTA MODEM))
3504* MDXTITLE (STRING (D(10) LISTA MODEM MULTIPLEXATI))
3505* MPXTITLE (STRING (D(10) LISTA MULTIPLEXER))
3506* TFGTITLE (STRING (D(10) LISTA TFG))
3507* RETETITLE (STRING (D(10) --- RETE TP DEL CNUCE --- PAG.))
3508* CDCITITLE (STRING (D(10) --- LISTA COLLEGAMENTI DIRETTI SU CONCEN TRATCRE (CDC) PAG.))
3509* CDITITLE (STRING (D(10) --- LISTA COLLEGAMENTI DIRETTI INTERURBANI (CDI) PAG.))
3510* CDUTITLE (STRING (D(10) --- LISTA COLLEGAMENTI DIRETTI URBANI (CDU) PAG.))
3511* COMTITLE (STRING (D(10) --- LISTA IMPIANTI COMMUTATI AL CNUCE (COM) PAG.))
3512* CTCTITLE (STRING (D(10) --- LISTA COLLEGAMENTI TRAMITE CONCENTRATORE (CTC) PAG.))

3513* FONTITLE (STRING (D(10) --- LISTA IMPIANTI FONIA (FON)
 PAG.))
 3514* ICCTITLE (STRING (D(10) --- LISTA IMPIANTI COMMUTATI SU CONCENTR
 ATORE (ICC)
 PAG.))
 3515* ICUTITLE (STRING (D(10) --- LISTA IMPIANTI COMMUTATI PRESSO UTEN
 TI (ICU)
 PAG.))
 3516* INTTITLE (STRING (D(10) --- LISTA IMPIANTI INTERNI (INT)
 PAG.))
 3517* TENTITLE (STRING (D(10) --- LISTA COLLEGAMENTI TEMPORANEI (TEM)
 PAG.))
 3518* BOLLATITLE (STRING (D(10) --- LISTA PATTUBE
 PAG.))
 4001* LISTA LCU (STRING (LIST /TITLE*LCUTITLE*,*FOCT*,*TPEHEAD*,*LCUHE
 AD*/*TPEID*,*LCUDATA*,OB C1,C2 WH C1 EQ LCU:))
 4002* PERIFERIA LCU (STRING (LI /TITLE,*FOOT*,*TPEHEAD*,*LCUHEAD*,*PER
 IFHEAD*/BY C0,*TPEID*,*LCUDATA*,*PERIFDATA*,OB C1101,C1102,C11
 03 WH C1 EQ LCU:))
 4003* LISTA TRM (STRING (LI/TITLE*TRMTITLE*,*FOOT*,*TPEHEAD*,*TRMHEAD*
 /*TPEID*,*TRMDATA*,OB C1,C2 WH C1 EQ TRM:))
 4004* LISTA LNE (STRING (LI/TITLE*LNETITLE*,*FOOT*,*TPEHEAD*,*LNEHEAD*
 /
 TPEID,*LNEDATA*,OB C1,C2 WH C1 EQ LNE:))
 4005* LISTA MDM (STRING (LI/TITLE*MDMTITLE*,*FOOT*,*TPEHEAD*,*MDMHEAD*
 /
 TPEID,*MDMDATA*,OB C1,C2 WH C1 EQ MDM:))
 4006* LISTA MPX (STRING (LI/REPEAT SUP,TITLE*MPXTITLE*,*FOOT*,*TPEHEA
 D*,*MPXHEAD*,*MPXCHINHEAD*,*MPXCHOUHEAD*/BY C0,*TPEID*,*MPXDATA*
 ,*MPXCHINDATA*,GE C1,C2 WH C1 EQ MPX:))
 4007* LISTA MDX (STRING (LI/TITLE*MDXTITLE*,*FOOT*,*TPEHEAD*,*MDXHEAD*
 ,*MDXCHINHEAD*,*MDXCHOUHEAD*/BY C0,*TPEID*,*MDXDATA*,*MDXCHINDAT
 A*,*MDXCHOUDATA*,OB C1,C2 WH C1 EQ MDX:))
 4008* LISTA TFG (STRING (LI/TITLE*TFGTITLE*,*FOOT*,*TPEHEAD*,*TFGHEAD*
 ,*TFGCHINHEAD*,*TFGCHOUHEAD*/BY C0,*TPEID*,*TFGDATA*,*TFGCHINDAT
 A*,*TFGCHOUDATA*,OB C1,C2 WH C1 EQ TFG:))
 4009* TFG= (STRING (LI/TITLE*TFGTITLE*,*FOOT*,*TPEHEAD*,*TFGHEAD*,*TFG
 CHINHEAD*,*TFGCHOUHEAD*/BY C0,*TPEID*,*TFGDATA*,*TFGCHINDATA*,*T
 FGCHOUDATA* WH C1 EQ TFG AND C2 EQ *1*:))
 4010* LNE= (STRING (LI/TITLE*LNETITLE*,*FOOT*,*TPEHEAD*,*LNEHEAD*/BY C
 0,*TPEID*,*LNEDATA*,OB C1,C2 WH C1 EQ LNE AND C2 EQ *1*:))
 4011* SITUAZIONE TERMINALI (STRING (LI/REPEAT SUP,TITLE*TRMTITLE*,*FO
 OT*,*TPEHEAD*,*STORIAHEAD*/*TPEID*,*STORIADATA*,OB C1,C2 WH C1 E
 Q TRM:))
 4012* CONNESSIONI DI (STRING (LI/TITLE,*TPEHEAD*,*CENTROHEAD*,*PERIFHE
 AD*/BY C0,*TPEID*,*CENTRODATA*,*PERIFDATA* WH C1 EQ *1* AND C2 E
 Q *2*:))
 4013* RETE TP (STRING (LI/TITLE*RETETITLE*,*FOOT*,*TPEHEAD*,*CENTROHEA
 D*,*PERIFHEAD*/BY C0,*TPEID*,*CENTRODATA*,*PERIFDATA*,OB C1,C2 W
 H C1 EXISTS:))
 4014* LISTACDC (STRING (LI/TITLE*CDCTITLE*,*FOOT*,*LINEEHEAD*/BY C0,C2
 ,C1403,C1401,C1404,OB C2 WH C1 EQ LNE AND ENTRY HAS C206 EQ CDC:
))
 4015* LISTACDI (STRING (LI/TITLE*CDITITLE*,*FOOT*,*LINEEHEAD*/BY C0,C2

,C1403,C1401,C1404,OB C2 WH C1 EQ LNE AND ENTRY HAS C206 EQ CDI:
))
4016* LISTACDU (STRING (LI/TITLE*CDUTITLE*,*FOOT*,*LINEEHEAD*/BY C0,C2
,C1403,C1401,C1404,OB C2 WH C1 EQ LNE AND ENTRY HAS C206 EQ CDU:
))
4017* LISTACOM (STRING (LI/TITLE*COMTITLE*,*FOCT*,*LINEEHEAD*/BY C0,C2
,C1403,C1401,C1404,OB C2 WH C1 EQ LNE AND ENTRY HAS C206 EQ COM:
))
4018* LISTACTC (STRING (LI/TITLE*CTCTITLE*,*FOOT*,*LINEEHEAD*/BY C0,C2
,C1403,C1401,C1404,OB C2 WH C1 EQ LNE AND ENTRY HAS C206 EQ CTC:
))
4019* LISTAFON (STRING (LI/TITLE*FONTITLE*,*FOCT*,*LINEEHEAD*/BY C0,C2
,C1403,C1401,C1404,OB C2 WH C1 EQ LNE AND ENTRY HAS C206 EQ FON:
))
4020* LISTAICC (STRING (LI/TITLE*ICCTITLE*,*FOOT*,*LINEEHEAD*/BY C0,C2
,C1403,C1401,C1404,OB C2 WH C1 EQ LNE AND ENTRY HAS C206 EQ ICC:
))
4021* LISTAICU (STRING (LI/TITLE*ICUTITLE*,*FOCT*,*LINEEHEAD*/BY C0,C2
,C1403,C1401,C1404,OB C2 WH C1 EQ LNE AND ENTRY HAS C206 EQ ICU:
))
4022* LISTAINT (STRING (LI/TITLE*INTTITLE*,*FOCT*,*LINEEHEAD*/BY C0,C2
,C1403,C1401,C1404,OB C2 WH C1 EQ LNE AND ENTRY HAS C206 EQ INT:
))
4023* LISTATEM (STRING (LI/TITLE*TEMTITLE*,*FOCT*,*LINEEHEAD*/BY C0,C2
,C1403,C1401,C1404,OB C2 WH C1 EQ LNE AND ENTRY HAS C206 EQ TEM:
))
4024* LISTA LINEE (STRING (*LISTACDU*: *LISTACTC*: *LISTACDC*: *LISTAINT*
: *LISTACDI*: *LISTAICC*: *LISTAICU*: *LISTACOM*: *LISTATEM*: *LISTAFO
N*:
))
4025* BOLLACDU (STRING (LI/REPEAT SUP,TITLE*CDUTITLE*,*FOOT*,*TPEHEAD
*,*BOLLAHEAD*/*TPEID*,*BOLLADATA*,OB C2,C1304 WH C1 EQ LNE AND E
NTRY HAS C206 EQ CDU AND C1304 CONTAINS SUFFIX 81:PR SUM C1302
WH SAME:))
4026* BOLLACTC (STRING (LI/REPEAT SUP,TITLE*CTCTITLE*,*FOOT*,*TPEHEAD
*,*BOLLAHEAD*/*TPEID*,*BOLLADATA*,OB C2,C1304 WH C1 EQ LNE AND E
NTRY HAS C206 EQ CTC AND C1304 CONTAINS SUFFIX 81:PR SUM C1302
WH SAME:))
4027* BOLLACDC (STRING (LI/REPEAT SUP,TITLE*CDCTITLE*,*FOOT*,*TPEHEAD
*,*BOLLAHEAD*/*TPEID*,*BOLLADATA*,OB C2,C1304 WH C1 EQ LNE AND E
NTRY HAS C206 EQ CDC AND C1304 CONTAINS SUFFIX 81:PR SUM C1302
WH SAME:))
4028* BOLLAINT (STRING (LI/REPEAT SUP,TITLE*INTTITLE*,*FOOT*,*TPEHEAD
*,*BOLLAHEAD*/*TPEID*,*BOLLADATA*,OB C2,C1304 WH C1 EQ LNE AND E
NTRY HAS C206 EQ INT AND C1304 CONTAINS SUFFIX 81:PR SUM C1302
WH SAME:))
4029* BOLLACDI (STRING (LI/REPEAT SUP,TITLE*CDITITLE*,*FOOT*,*TPEHEAD
*,*BOLLAHEAD*/*TPEID*,*BOLLADATA*,OB C2,C1304 WH C1 EQ LNE AND E
NTRY HAS C206 EQ CDI AND C1304 CONTAINS SUFFIX 81:PR SUM C1302
WH SAME:))
4030* BOLLAICC (STRING (LI/REPEAT SUP,TITLE*ICCTITLE*,*FOOT*,*TPEHEAD

Appendice C - Un esempio di sessione

```
profile
CP LINK CMSMU 192 192 RR
ACC 192 D
'192' REPLACES ' D (192) '
D (192) R/O
MSS ATT TPDB 333 TPDB
DIGITARE LIST PER UTILIZZARE LE FUNZIONI DI LISTA
DIGITARE PLEX PER UTILIZZARE LE ALTRE FUNZIONI
DIGITARE EXIT PER USCIRE
list
EXEC TP
STARTING SESSION -----
FI LCU DISK TP LCU A ( PERM
FI MPX DISK TP MPX A ( PERM
FI MDX DISK TP MDX A ( PERM
FI MDM DISK TP MDM A ( PERM
FI TRM DISK TP TRM A ( PERM
FI TFG DISK TP TFG A ( PERM
FI LNE DISK TP LNE A ( PERM
FI STRINGHE DISK TP LIST A ( PERM
CP SP E CL 2
FI PRINTER PRINTER ( RECFM FBA LRECL 133 BLOCK 1330 PERM
. . . C O N T I N U A . . .
R; T=0.05/0.16 11:07:20
'333' REPLACES ' A (333) '
R; T=0.01/0.02 11:07:22
'191' REPLACES ' B (191) '
B (191) R/O
R; T=0.01/0.01 11:07:22
EXECUTION BEGINS...
S2K0104/00- SYSTEM 2000 INITIALIZATION PARAMETERS FOLLOW -
S2K0127/01- WARNING- STAF ERROR TRAPPING NOT ENABLED-
S2K0212/01- SYSTEM 2000 RELEASE 10.1-XX -
04/29/83 11:07:41 BEGIN SYSTEM 2000 - RELEASF 10.1-XX
---
user,siba:dbn is tp:
USER,SIBA:DBN IS TP:
-556- OPENED.....TP
10 41 04/28/1983 09:53:50
---
```

comando ==> *listacom*:

output ==> --- LISTA IMPIANTI COMMUTATI AL CNUCE (COM)

03/15/1983

* NUMERO DI TD DISLOCAZIONE DELLA LINEA

* 050/40118	Pisa	Servizio Guasti
* 050/41387	Pisa	Operatore TP
* 050/501043	Pisa	Centrale GTEX
* 050/501167	Pisa	Centrale GTEX
* 050/501215	Pisa	Centrale GTEX
* 050/501301	Pisa	Centrale GTEX
.	.	.
.	.	.
.	.	.
* 050/502709	Pisa	S/S 300 bps

--- FINE PAGINA ---

comando ==> *listactc*:

output ==> --- LISTA COLLEGAMENTI TRAMITE CONCENTRATORE (CTC)

03/15/1983

* NUMERO DI TD DISLOCAZIONE DELLA LINEA

* 700010/01	Padova	Area Ricerca CNR	Pisa	CNUCE
* 700010/02	Firenze	Ist. Matem. U. Dini	Pisa	CNUCE
* 700010/03	Roma	Ist. Applicaz. Calc.	Pisa	CNUCE
* 700010/04	Milano	Staz. Remota CNUCE	Pisa	CNUCE
.
.
.
.
* 700010/12	Bologna	Fac. Ingegneria	Pisa	CNUCE

--- FINE PAGINA ---

exit:

EXIT:

11:07:55 04/29/83 END SYSTEM 2000 - RELEASE 10.1-XX R;

Rispondendo alla domanda iniziale con 'PLEX', l'applicazione guida l'operatore, tramite la tecnica dei menu nelle azioni da farsi per eseguire le funzioni previste dall'applicazione stessa.