NAME

ascii – map of ASCII character set

SYNOPSIS

cat /usr/pub/ascii

DESCRIPTION

Ascii is a map of the ASCII character set, to be printed as needed. It contains:

000	nul	001	soh	002	stx	003	etx	004	eot	005	enq	006	ack	007	bel
010	bs	011	ht	012	nl	013	vt	014	np	015	cr	016	so	017	si
020	dle	021	dc1	022	dc2	023	dc3	024	dc4	025	nak	026	syn	027	etb
030	can	031	em	032	sub	033	esc	034	fs	035	gs	036	rs	037	us
040	sp	041	!	042	"	043	#	044	$	045	%	046	&	047	´
050	(	051	)	052	*	053	+	054	,	055	–	056	.	057	/
060	0	061	1	062	2	063	3	064	4	065	5	066	6	067	7
070	8	071	9	072	:	073	;	074	<	075	=	076	>	077	?
100	@	101	A	102	B	103	C	104	D	105	E	106	F	107	G
110	H	111	I	112	J	113	K	114	L	115	M	116	N	117	O
120	P	121	Q	122	R	123	S	124	T	125	U	126	V	127	W
130	X	131	Y	132	Z	133	[	134	\	135	]	136	^	137	_
140	`	141	a	142	b	143	c	144	d	145	e	146	f	147	g
150	h	151	i	152	j	153	k	154	l	155	m	156	n	157	o
160	p	161	q	162	r	163	s	164	t	165	u	166	v	167	w
170	x	171	y	172	z	173	{	174	|	175	}	176	~	177	del

FILES

found in /usr/pub

NAME

dpd – spawn data phone daemon

SYNOPSIS

/etc/dpd

DESCRIPTION

Dpd is the 201 data phone daemon. It is designed to submit jobs to the Honeywell 6070 computer via the GRTS interface.

Dpd uses the directory */usr/dpd*. The file *lock* in that directory is used to prevent two daemons from becoming active. After the daemon has successfully set the lock, it forks and the main path exits, thus spawning the daemon. The directory is scanned for files beginning with **df.** Each such file is submitted as a job. Each line of a job file must begin with a key character to specify what to do with the remainder of the line.

S directs *dpd* to generate a unique snumb card. This card is generated by incrementing the first word of the file */usr/dpd/snumb* and converting that to three-digit octal concatenated with the station ID.

L specifies that the remainder of the line is to be sent as a literal.

B specifies that the rest of the line is a file name. That file is to be sent as binary cards.

F is the same as **B** except a form feed is prepended to the file.

U specifies that the rest of the line is a file name. After the job has been transmitted, the file is unlinked.

Any error encountered will cause the daemon to drop the call, wait up to 20 minutes and start over. This means that an improperly constructed *df* file may cause the same job to be submitted every 20 minutes.

While waiting, the daemon checks to see that the *lock* file still exists. If it is gone, the daemon will exit.

FILES

/dev/dn0, /dev/dp0, /usr/dpd/*

SEE ALSO

opr(I)

NAME

getty – set typewriter mode

SYNOPSIS

/etc/getty

DESCRIPTION

Getty is invoked by *init (VII)* immediately after a typewriter is opened following a dial-up. The user's login name is read and the login(I) command is called with this name as an argument. While reading this name *getty* attempts to adapt the system to the speed and type of terminal being used.

Getty initially sets the speed of the interface to 150 baud, specifies that raw mode is to be used (break on every character), that echo is to be suppressed, and either parity allowed. It types the ''login:'' message (which includes the characters which put the 37 Teletype terminal into full-duplex and unlock its keyboard). Then the user's name is read, a character at a time. If a null character is received, it is assumed to be the result of the user pushing the ''break'' (''interrupt'') key. The speed is then changed to 300 baud and the ''login:'' is typed again, this time with the appropriate sequence which puts a GE TermiNet 300 into full-duplex. This sequence is acceptable to other 300 baud terminals also. If a subsequent null character is received, the speed is changed back to 150 baud.

The user's name is terminated by a new-line or carriage-return character. The latter results in the system being set to to treat carriage returns appropriately (see stty(II)).

The user's name is scanned to see if it contains any lower-case alphabetic characters; if not, and if the name is nonempty, the system is told to map any future upper-case characters into the corresponding lower-case characters. Thus UNIX is usable from upper-case-only terminals.

Finally, login is called with the user's name as argument.

SEE ALSO

init(VII), login(I), stty(II)

NAME

glob – generate command arguments

SYNOPSIS

/etc/glob command [arguments]

DESCRIPTION

Glob is used to expand arguments to the shell containing ‘‘*’’, ‘‘[’’, or ‘‘?’’. It is passed the argument list containing the metacharacters; *glob* expands the list and calls the indicated command. The actions of *glob* are detailed in the Shell writeup.

SEE

sh(I)

BUGS

Glob gives the ‘‘No match’’ diagnostic only if no arguments at all result. This is never the case if there is any argument without a metacharacter.

NAME

greek – graphics for extended ascii type-box

SYNOPSIS

cat /usr/pub/greek

DESCRIPTION

Greek gives the mapping from ascii to the ‘‘shift out’’ graphics in effect between SO and SI on model 37 Teletypes with a 128-character type-box. It contains:

alpha	α	A	beta	β	B	gamma	γ	\
GAMMA	Γ	G	delta	Δ	D	DELTA	Δ	W
epsilon	ε	S	zeta	ζ	Q	eta	η	N
theta	θ	T	THETA	Θ	O	lambda	λ	L
LAMBDA	Λ	E	mu	μ	M	nu	ν	@
xi	ξ	X	pi	π	J	PI	Π	P
rho	ρ	K	sigma	σ	Y	SIGMA	Σ	R
tau	τ	I	phi	ϕ	U	PHI	Φ	F
psi	ψ	V	PSI	Ψ	H	omega	ω	C
OMEGA	Ω	Z	nabla	∇	[	not	¬	_
partial	∂	]	integral	∫	^			

SEE ALSO

ascii (VII)

NAME

init – process control initialization

SYNOPSIS

/etc/init

DESCRIPTION

Init is invoked inside UNIX as the last step in the boot procedure. Generally its role is to create a process for each typewriter on which a user may log in.

First, *init* checks to see if the console switches contain 173030. (This number is likely to vary between systems.) If so, the console typewriter *tty* is opened for reading and writing and the shell is invoked immediately. This feature is used to bring up a single-user system. When the system is brought up in this way, the *getty* and *login* routines mentioned below and described elsewhere are not needed.

Otherwise, *init* invokes a Shell, with input taken from the file */etc/rc*. This command file performs housekeeping like removing temporary files, mounting file systems, and starting the dataphone daemon.

Then *init* forks several times to create a process for each typewriter mentioned in an internal table. Each of these processes opens the appropriate typewriter for reading and writing. These channels thus receive file descriptors 0 and 1, the standard input and output. Opening the typewriter will usually involve a delay, since the *open* is not completed until someone is dialled up and carrier established on the channel. Then the process executes the program */etc/getty* (q.v.). *Getty* will read the user's name and invoke *login* (q.v.) to log in the user and execute the shell.

Ultimately the shell will terminate because of an end-of-file either typed explicitly or generated as a result of hanging up. The main path of *init*, which has been waiting for such an event, wakes up and removes the appropriate entry from the file *utmp*, which records current users, and makes an entry in *wtmp*, which maintains a history of logins and logouts. Then the appropriate typewriter is reopened and *getty* is reinvoked.

FILES

/dev/tty, /dev/tty?, /tmp/utmp, /tmp/wtmp,

SEE ALSO

login(I), getty(VII), sh(I)

NAME

msh – mini-shell

SYNOPSIS

/etc/msh

DESCRIPTION

Msh is a heavily simplified version of the Shell. It reads one line from the standard input file, interprets it as a command, and calls the command.

The mini-shell supports few of the advanced features of the Shell; none of the following characters is special:

> < $ \ ; & | ^

However, ‘‘*’’, ‘‘[’’, and ‘‘?’’ are recognized and *glob* is called. The main use of *msh* is to provide a command-executing facility for various interactive sub-systems.

SEE ALSO

sh(I), glob(VII)

NAME

tabs – set tab stops

SYNOPSIS

cat /usr/pub/tabs

DESCRIPTION

When printed on a suitable terminal, this file will set tab stops every 8 columns. Suitable terminals include the Teletype model 37 and the GE TermiNet 300.

These tab stop settings are desirable because UNIX assumes them in calculating delays.

NAME

tmheader – TM cover sheet

SYNOPSIS

ed /usr/pub/tmheader

DESCRIPTION

/usr/pub/tmheader contains a prototype for making a *troff(I)* formatted cover sheet for a technical memorandum. Parameters to be filled in by the user are marked by self-explanatory names beginning with ‘‘---’’.

BUGS

God help you on two-page abstracts. Try to write less.

NAME

vs – voice synthesizer code

DESCRIPTION

The octal codes below are understood by the Votrax® voice synthesizer. Inflection and phonemes are or-ed together. The mnemonics in the first column are used by *speak* (I); the upper case mnemonics are used by the manufacturer.

0	300	4–strong inflection
1	200	3
2	100	2
3	000	1–weak inflection
a0	033	AH–c**o**ntact
a1	052	AH1–c**o**nnect
aw	002	AW–l**aw**(,l,u2,aw)
au	054	AW1–f**au**lt
ae	021	AE–c**a**t
ea	020	AE1–**a**ntenna
ai	037	A–n**a**me(,n,ai,y0,m)
aj	071	A1–n**a**mely
e0	004	EH–m**e**t **e**nter
e1	076	EH1–s**e**ven
e2	077	EH2–sev**e**n
er	005	ER–weath**er**
eu	073	OOH–G**oe**the chev**eu**x
eh	067	EHH–l**e** ch**e**veux
y0	023	EE–thr**ee**
y1	026	Y–sixt**y**
y2	035	Y1–**y**es
ay	036	AY–ma**y**
i0	030	I–s**i**x
i1	064	I1–**i**nept **i**nside
i2	065	I2–stat**i**c
iy	066	IY–cr**y**(,k,r,a0,iy)
ie	003	IE–z**e**ro
ih	072	IH–stat**i**on
o0	031	O–**o**nly n**o**
o1	012	O1–hell**o**
o2	013	O2–n**o**tice
ou	051	OO1–g**oo**d sh**ou**ld
oo	050	OO–l**oo**k
u0	014	UH–b**u**t
u1	015	UH1–**u**ncle
u2	016	UH2–stirr**u**p
u3	034	UH3–app_le ab_le
yu	027	U–**u**se
iu	010	U1–**u**nite(,y1,iu,...)
ju	011	IU–n**ew**
b	061	B
d	041	D
f	042	F
g	043	G
h	044	H
k	046	K
l	047	L
m	063	M
n	062	N
p	032	P
q	075	Q
r	024	R
s	040	S
t	025	T
v	060	V
w	022	W
z	055	Z
sh	056	SH–**sh**ow **sh**ip
zh	070	ZH–plea**s**ure
j	045	J–edge
ch	057	CH–bat**ch**
th	006	TH–**th**in
dh	007	THV–**th**en
ng	053	NG–lo**ng** i**n**k
–0	017	PA2–long pause
–1	001	PA1
–2	074	PA0–short pause

SEE ALSO

speak(I), vs(IV)