Wed Nov 28 13:26:47 1979

SHORT GUIDE TO UNIX PROLOG IMPLEMENTATION

TO UNI X PROLOG IMPLEMENTATION

an La delandaria a

Weight

Her 199**0 (1996)** - 197

a d**ahaye**ka kalan dala

Marken dillered .

Contraction in the second 20.004

1.0000

🗰 olgo – o ngalimologija

nin sainin san

🖗 i sa na kalimena ka al

🗰 - haay 📷 🏭 Malaka hiri ya Kutoka na sa sa sa sa katala ka ka

in waan

(#k)

C.S.Mellish, 25 October 1979

1. Introduction

ويقرد الدا

------This wis a very short note to help anybody who might wish to alter or make maximum additions to UNIX PROLOG. A fuller description may appear later, but for now this summary is all that is available. The description applies to version NU4 meronomic contraction of the system, although many details are more or less independent of the implementation and of understanding whatever obscurities the code and its comments may have.

> Some of the design decisions now look rather strange in hindsight, but this is not the place for justifications or suggestions for improvements.

2. Storage of Programs

The PROLOG system is an interpreter, and so the storage of user programs involves little more than representing in a linear form the syntactic structure warmen of the clauses.

A fundamental component of the representation system is the representation of we a word in the hash table (which extends from • address of a dictionary entry for an atom. If the word that is obtained • ----- contains zero, this means that the atom has not been previously encountered. In A second seco second point to that. On the other hand, if the word obtained is non-zero, it is necessary to check whether the dictionary entry pointed to is the one for the characters stored in the dictionary entry) If the entry is for some other atom, and set some success there. This process continues until the system has finally a present of the distinguish of the table that points to the distionary entry for the atom (a me entry being created if ever a zero word is encountered). The position of

SHORT

impl

Int.	and an end of process of the		1	
			4	
ingen in inden son open open open open open open open op			• • • • • • • • • • • • • • • • • • •	
 International and the second seco	Here and the second		• į	
namen - in defense and in president was LOW	addr:	Addr of first clause		
en an eine state and the second s	An Martinian and Grand and a state of the		• • • • • • • • • • • • • • • • • • •	
	ana suidean ann a' suideach an an a' suideach a' suideach a' suideach a' suideach a' suideach a' suideach a' su	After Before	and the second	
The second se	en e	Precedence Precedence		
and the second	en de skrigt de skrivere en de ser de ser se		 A second state 	
/ 1998 - Nach I, Maria Maria (1997-1994) (1973) An Andrew (1974)	Marine Carlos and	1st char Length		-
			 Control test and the second sec	
		3rd char 2nd char		
and the second se	an 'n den gebruik weten en s		an an an ann	
na se ner te norman en anteres en anteres estas	para presenta de constru		a start a start	
🗰 🖂 an	enager Scagar Anna Galerán (Anna Anna	, 	·	
e i Hanaska sa kasakanan manakanan g	h addr:	0 Inth char		
n en	a ta pipalan ay a si san si		i i standarda en la compañía de la c	
🕷 🔍 AMARAN MARKAT ANARAN SA A SA S				
	g ang tao ang	Figure 1: Dictic	onary Entry for Atom	Fisar
ne internet de la composition				
s0	on ountil a	zero pointer is f	found. The next two butes give precedence	
inf	ormation about	, how the atom behaves	s as an operator. They usually contain	·
	o, but can be	filled by the rout in	e OP. The 'length' stored in a dictionary	
ent	ry is the num	per of characters in t	he name (n, say) +1, all rounded up to a	
mill	tiple of 2.	If n is even, the o	fictionary entry is padded out with a null	·
but	e at the end.			
an a	ang sengagi si k			
n an	be neoresent.at	ion of a clause is il	lustrated in figure 2.	
nin - Andrea	1993 - 1997 - 1997 - 1998 - 1997 - 1998 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 -		renormaran varit vativei van van tit i i van bart vat nanona.	
	Alexandro and a state of the st	-	• E State	
	ı addr:	Addr of pext clause		
	ana an		 A state of the sta	
Martin Malanamatika dalah di kurana serian kenarata kasar dalam dalam dalam dalam dalam dalam dalam dalam dalam	Den Fait - Marsha	Number of Variables	ton and a second secon	
n an	remulais of the second second		· · · · · · · · · · · · · · · · · · ·	
 A second sec second second sec	an care i andro a care a c	i se an anna an an an an Anna a	er Bern Turbular (1.16)	
	a na senera parte da serie da la calendaria da la calendaria da senera da serie da serie da serie da serie da s	Head of clause	 Control of the second se	
 Construction of the second s Second second seco second second sec	nnen er er en sammen som er		 Markovské vyvolství stalovatel stalov Stalovatel stalovatel stal	
With Comparison Difference Control			ne terre and an and a second sec	
	na na serie de la companya de la com	l tekanal		
	and a second	I IBC BOUL		
name in the experimental constraints of the second states of the second	n en an		• • •	
n an	an a			
an an an ann an ann an ann an an an an a	n namatistaan si jagi ni min kenagan si	En le nationa de la seconda de la seconda En la seconda de la seconda	No. 19 No. 19 N	
1999 - Anna Anna Anna Anna Anna Anna Anna An	anna, ann ann ann an ann an ann an ann an	E de la construcción de marcina de la construcción de la construcción de la construcción de la construcción de		
ande population de la Andrea de Calendaria de Calendaria de Calendaria de Calendaria de Calendaria de Calendaria La composición de Calendaria	engelen die Prinzipense en eine eine Angelen in Berefit gehieren in eine eine		• • • • • • • • • • • • • • • • • • •	
al - La mana an a ang kataon na kataon na sara-	and provide the state of the second provide state of	le c u nu nu n u de la composition de la compo		
r 🗰 – 11. – Alifitit A. Ingele A. Angeliking (Proj. 1997) – Alifitit I. 🖌 🚺	ih addr:	i nth soal		
	Ng M ^a nanana (1999), na tarta ang ang ang ang ang ang ang ang ang an	n an		
$(10)^{-1} e^{-\frac{1}{2} \left(\frac{1}{2} + \frac{1}{2} +$	Red Red Rev Market Control of Control of State	And a second		
tinn altrahung ng Jakasa sakahini kang sakahini sa sa	u ars đu đế			
and Declarge a filling a set faint a constant of a set of a set of the set of	n na séndé	Fisure 2: Clau	use Representation	Fisur
and a standard a standard a standard a standard a s An an	n na el nate des elementas y constituis na constituis des elementas y constituis na constituis de	Figure 2: Clau	use Representation	Fisur
an arain an	nouche Reférence (1990) - in neuro Reference Ref rict - in COMPORE	Figure 2: Clau	use Representation	Fisur
Contraction of the second s	the head term	Figure 2: Claunts of the representat and the goals in the	use Representation tions of clauses are the representations body of the clause. Each one of these may	Fisur
constant and the second s	noutes Senation Componen Lhe head term	Figure 2: Claunts of the representate and the goals in the	use Representation ions of clauses are the representations body of the clause. Each one of these may 2	Fisur
	nautor Haddaatu (2000) oo ee oo 2000 ay Ka ma i A a GOMPONE Lhe head term Matter	Figure 2: Claunts of the representate and the goals in the	use Representation ions of clauses are the representations body of the clause. Each one of these may 2 area	Fisur
Alexandre and a set of the second secon	nauche Maries Statut (2000) (Connector Somain - Componer Lhe chead term Maries Mead term	Figure 2: Claunts of the representate and the goals in the	use Representation tions of clauses are the representations body of the clause. Each one of these may 2 area	Fisur
	nauche Maine (1990) (Constant So main - Componer Lhe - head term Maine (1990)	Figure 2: Claunts of the representat and the goals in the	use Representation tions of clauses are the representations body of the clause. Each one of these may 2	Fisur

----'

20 L 854 ---- - Andrew State Physics and the

represented. The representation of any goal in a clause happens to begin with a • Commence word containing an odd number whereas each atom entry or clause begins with an even value. Hence the end of a clause does not have to be specially marked.

•С

🗰 Co Astronotico de La co

. The second second

🗰 🛫 👬 🗰 👘 🖓 Start Barrier (Start) - Start

and the second second

🗰 🦕 🙀 and a construction of the second seco

🗰 o 🗝 saanaadaa iyo in Coolin ahaasaa da ahaasaa ka baayaa da ahaa iyo

🕷 🚕 dheller damid de marge sous a celado d

🗰 📩 a forgangengen og statige som som frem om at d

alle a deserved to the second second

• C . served to the server of all spaces and space **60**

🗰 la sina**ntan** mangan

ing a state in the .

a desta d Recessiones desta dest met

An example of the second s

-

ні :

a sangin isa kite

High addr:

 $-\Delta_{i}\delta_{i}=-\frac{1}{2}(\lambda_{i}-\lambda_{i})+\frac{1}{2}(\lambda_{i}-\lambda$

🗰 🦦 Antonio de la contra de

" Communications of the terms that are the head and the soals of clauses modeline entropy of the following rules:

For an integer - the representation takes the form of 1 word in the

For a variable - the representation takes the form of 1 word • _____ I <= n.<= number of vars).

m menession and an anonymous variable (a variable that only occurs once in a • ______ character clause and hence does not have to be allocated any space) - a single • Variable Containing 0.

🗝 🔬 🚥 where we want the second station is a single word with the format . where the atom number.

* Some second a complex term - this is represented in a sequence of words; 🗰 🛶 🛶 🗤 where the second maximum word nepresenting the functor of the term and continues with the e complex terms). The format of a word 🗰 🛶 alina ing manganan ing mangani na d

An example of the representation of a term is given in figure 3.

ii	State State and the
10011001101000111	(3,f)
000000000000000000000000000000000000000	(1)
000000000000000000000000000000000000000	(var 1)
0010010001011111	(2, g)
000000000000000000000000000000000000000	(var 1)
000000000000000000000000000000000000000	(anon)

Figure 3: Skeleton for f(1,X,g(X,Y))

Figure

------Examples of atom entries and clauses can be seen at the end of the PROLOG metanement system code. Even where they have special system-defined properties (ie when 3

represented as above. If a predicate is associated with a system function, the address of the routine must be placed in the word before the start of a dummy clause for the predicate. The clause must also appear before the label SFSTART (and only such clauses can appear there). Then when the head of the clause is matched, instead of looking at the body of the clause the system calls the special routine.

3. Storage of Data in Execution

-

a constant

West montilelised

198 * ıń. jà là

ŧ.

i#

Rill(

MR-1.1

int is written ----

The standy

🖷 Sanda (Si 🎆 magan sa sa 🐞 . Sangingin - S

n Latitite course

Weise and the

• 2****

🐝 🗸 💏 🖓 de la secte de la composición de la CARACTER CONTRACTOR CONTRACTOR

🗰 - E**rikan**i (miran-1961). - 19

And the part of the

· Longe

20**00000** s Case

🙀 - ng San Siri

* ~** mi Constanti

NIN THE REAL PROPERTY. ti Casa

sections of the section of the secti

🗰 🖓 👘 👘 👘 👘

🙀 a 🗰 🎼 de la companya de la compa

We see the set of the

🙀 🕻 👬 🗰 🖓 🖓 👘 🖓

We wanted to the second second

and Support and Support

MAD OF STREET

nin ainmaighe a

nikai sangerik Å

1

1

He Sundaningensieners un

alle Sandalaine contra

and the second second

N. Cano

A Construction of the second s

> - When the user runs a program, he assigns values to variables corresponding to invocations of clauses. When a clause is invoked, the system assigns a block of variable is 'undefined', which is represented by the value 0. During the execution, the value may change to one of the following:

where variables are unified. The later instantiation of the second one must result with the same value being associated with the first. The effectively "To set the value, look in this other place".

- - An atomic item (inteser or atom). In this case the representation of the item (being the same as in a skeleton) is put in the word.

- A complex term. Complex terms created during execution (DS's) are placed in the slobal stack. If a variable is siven such a value, a ----pointer to the structure is placed in the variable's word. A DS is effectively a copy of a skeleton with values like those we are

4. Core Allocation

The organisation of core allocation has been designed to make the most of the (rather inappropriate) facilities that UNIX offers in this area. The system attempts never to demand more from the operating system than it is likely to need. First of all, the overall layout of core is illustrated in fisure 4. Ał. any point, the PROLOG system must be able to access the area between the lowest address and GTOP, as well as the area between LTOP and TOP. The former of these is in general allocated as UNIX data area, with DATLIM giving the address of the highest location that can currently be accessed. The latter is in seneral allocated as UNIX stack area, with STLIM siving the address of the next word below this area. When the stack area attempts to expand over 4K words further expansion is prevented and the data area expands instead to meet it. The flas CONNECTED indicates whether this condition holds. Although the PROLOG system relinguishes control of data space that it does not immediately need (through TRIMCORE), it is unfortunately not possible to do this with stack 4

Lowest Address: HASHSTART: Sustem code Hash table for atoms HASHEND: System clauses and atoms User's clauses and atoms (heap) FREESTART: HTOP---> Free area above heap 20 20 20 20 10 HLIMIT---> Previous slobal stack areas *<i>и и и и и и и* GSTART--> Current soal clause and DS A state of the second GBOT---> Current slobal stack GTOP---> Free area between stacks H H H H H H H ser dia 2.862° Meridia Current local stack LTOP---> Previous local stack areas LBOT--> Top of core TOP--->

• Ĉ

Contraction of the second

in section in the section of the sec

nan Cheven

• C....

*Č***

de la station

n Charles in t

1994 (1997) - 1993 (1996) (1997) 1994 (1997) - 1997 (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (

а. **С**

1. 1

and consideration

dia mpinina

Mannan M

ALL CONTRACTOR

-Come

an and and a constant

1996 C. 1983 P. 1996

C

C

Card and the second states of the

1. A.

"Ĉ

1. **19** 19 19 19 19 19

an seattain sea

6

Ö.

Figure 4: Core Layout

space. Since it is useful to keep the SP resister to handle subroutine calls within the interpreter, SP is not used as a pointer to the stacks at the top of core except when more space needs to be allocated there. In seneral, SP points inside a special workspace area extending downwards from WSTART.

Because of the facility to do nested 'consult's and 'breaks', it is possible that the user may have several different execution environments active at the same time. The main parameters of the current environment are to be found between ENVSTART and ENVEND. The parameters for previous environments are kept on the slobal stack.

areas: which in the space that the system uses, there are basically three main data areas:

1. The 'heap'. This is not really a heap, because it is not sarbase collected. Instead it is a stack containing all the permanent clauses and atom dictionary entries. This area starts at FREESTART and its top is siven by HTOP. The expansion is limited to below HLIMIT, which is where the slobal stacks start. If expansion further is attempted, the slobal stacks are automatically moved up to make space. Anything placed on the heap must start with a word with an even value (because there are no explicit clause terminators). Note that the function to check the availability of space on the heap (HALLOC) does not update HTOP. This is so that it is possible to put values there but to postpone making them permanent until the last

minute. In this way one can avoid permanently messing up the heap if an error occurs half way through putting something on.

2. The slobal stack areas. There is one slobal stack for each active user execution. The first one's start is siven by HLIMIT and the

Fisdre

7

 $\tilde{\langle}$

-----current one's by GBOT. The top of the current one is given by GTOP. The slobal stack is used to store terms constructed during the - execution (referred to in many comments as DS's). In addition, the DS and clause associated with the current user soal are kept just below values must be reset on backtracking. Trail entries are interspersed with DS's on the slobal stack; they can be told apart because a trail entry is a single address (even value) whereas a DS starts with an odd value. If the stack is moved up, the pointers GBOT, GSTART, OPSTART, HLIMIT and GTOP are updated, as are any pointers into the see slobal stack from itself or the local stack. No other pointers are updated. Putting anything on the global stack must be preceded by an appropriate call of GALLOC.

execution. The local stacks expand downwards, whereas the slobal where stacks and heaps expand upwards. The first local stack begins at the address given in TOP. The current one begins at the value of LBOT and - extends to the value of LTOP. Local stacks are used to contain administrative information about the user's execution as well as we cells for variable values. These are allocated in groups called stack --------frames. Puttime anything on the local stack must be preceded by an appropriate call of LALLOC. Note that the part of the stack used ----during the satisfaction of a goal will be reclaimed if this succeeds determinately.

5. Keeping track of Execution

....**C**

1

Ser Crasses and a series

top Versialist man dia mangatish no sojen sol

and the stand of the state of the

Reference and an and

w California and a second

AND A STATE OF A STATE

itali onta antico a constructione de la construcción de la construcción de la construcción de la construcción d

Manager Anter Anter Statistics

elle line o colarette o cito suc

TREAMPERS IN AND A SHARE IN THE

sines of the second of the

age inder opper-

And State a

🐝 🖳 da sa katala kata

NET THE REPORT OF THE PARTY OF THE PARTY OF

A THE MENT OF STREET, SAME AND A STREET, SAME

HALL STATE BORNER STATE STATE

A Constant Constant State of the second state of the second state of the second state of the second state of the

A DESCRIPTION OF THE PARTY OF T

and the second state of the second second

HAR - HAR HAR - 12-22

inger a

waish -

estate de

sign.

an **C**anasasa -

WHAL CONTRACTORS

🗰 alata 👬 de calendaria e constante constante constante e constante e consta

> The code for the main loop of the interpreter (from CONTINUE to GOON) is fairly well commented, but the main structure can be summarised here.

 \sim Given the skeleton for a goal to be satisfied (at GOTCALL) the system obtains generation can be atom which acts as its functor and finds the first clause with that atom as 2 stack for a new invocation of the clause. It starts off by recording -administrative information and then (at ENTER) allocates space for the - variables of the clause. It then invokes the unification routine (UNIFY). One -arsument to this is a pointer to the skeleton that mentioned this soal, 2 the head of the clause, together with the address of the new stack frame generation (siving the values of the variables for the new invocation).

6

---- If unification succeeds, (now at CONTINUE) it must first be checked to see if a - system clause has been activated. If not, the search for the next goal can take place immediately. The first place to look is after the head of the new Э. clause (now at TRYCALL). If there is a soal there, that is fine; otherwise we -have completed the current soal. In this case (now at RETURN) we must look back www.americlancience.to the soal that this was a subsoal of, and so on until a remaining unsatisfied soal can be found. If this process soes back right to the original user soal, www.www.www.www.www.have.been successful (SUCCESS).

)

Э

2

2

)

)

-----If unification fails, or there are no clauses for the predicate, backtracking takes place (FAIL). After resetting the values of the variables given in the **Communications that is the system looks for the last choice point (The address of its stack** frame is given in NB). If there is none, the user goal cannot be satisfied (FAILURE). Otherwise NB is reset to the previous choice point and a new clause is picked for the place where the choice was made. (Now back to GOTCLAUSE).

> How does the execution process start (at EXECUTE)? The soal that the user sives is 'asserted' as the body of a clause for the predicate '' (the atom with an empty name). That is, the atom entry for "? (at EDUMMY) is made to point to a dummy clause at the start of the slobal stack. The system then sets the address (DCALL) of a dummy soal of " and starts the interpreter cycle at GOTCALL.

> The details of the interpreter loop can only really be understood if the format of stack frames is explained. Figure 5 does this.

6. Unification

r 🗰 💭 en en se en Bresselve service service

A REAL PROPERTY AND A REAL

the management of the second

ALC: MARKET PROVIDE A CONTRACTOR

and the second

Mar and the second of

n **China an an Anna an**

THE TOTAL STREET THE STATE STATES

and the mean and the second

inin ini Angelia

Main 他的書 Charles Charles

The set the state of the second

Oha - Bashiaan too goo moo com

NAME OF THE PARTY Contraction Marcon Incon-

.........

🗰 🗣 racantes 👘 Martin Participation and the second n ang an s

ana santa 5.00 K

Matter to a statistica and the THERE S. Maria

an Balanser

the state of the second se

NEW COMPANY Sec. And Constant

-

AN INCOMENTATION CONTRACTOR

Contractor and the second

sa C

- During the execution of a user's program, there are two ways in which

instantiations of the skeletons in his clauses can be characterised. Firstly, siving the address of a skeleton plus the address of a stack frame (normally the *"*base" address) gives enough information to specify the term denoted, 7 because the skeleton gives the basic structure and the stack frame gives the values of the variables that fit into that structure. Secondly, giving the address of a concrete DS is also a complete specification. The basic)) unification routines make use of the similarities in the representation of skeletons and DS's and work for both types of representation. The only difference is that if the number of a variable is encountered (which only Э. happens in a skeleton) the appropriate environment (stack frame) must be available, so that the value of the variable can be looked up. The locations E, E1 and E2 are used to hold currently used environment pointers.

The routine IDENT is fundamental for unification. Its function is to interpret the data representations. IDENT is called to identify the type of a 3 term-and also to return an appropriate value result. It can be applied to a variable cell or to part of a skeleton or DS. If it is used on a skeleton, the environment pointer E must be set appropriately. The possible results that Э, IDENT can return are given in fig 6. Most of what IDENT does is just following pointers, decoding the last two bits of words and looking up the values of

7

Low addr:

			** *** *** *** ***	4
: Cell	for	var	n	ł
[ł
1 				ł
1				ł
!				ł
Cell	for	var	1	1
!				ł
Addr	of (laus	5C	ł
! heine	1 1157	əd		1

Ĵ.

)

2

Э.

2

2)

	Addr of call of this soal	
n an	Base of frame for parent soal	<pre>({== Base of stack frame (FRAME)</pre>
	Base of frame for last choice point Value of GTOP	<pre>> Only > present > for</pre>

- - For an anonymous variable (a variable that only occurs once in a clause and hence does not have to be allocated any space) - a single word containing 0.

- For an atom - the representation is a single word with the format \sim (4 bits)(10 bits)11, where the first 4 bits are zero and the 10 bits are the atom number.

- For a complex term - this is represented in a sequence of words, siving the prefix Polish structure of the term. Thus it begins with a word representing the functor of the term and continues with the course, themselves be complex terms). The format of a word we represent in a functor is (4 bits)(10 bits)11, where the 4 bits aive

An example of the representation of a term is siven in fisure 3.

na servera de la composición de la comp

State for the state of the stat No. A Continued and a second A Street and the second se

and a state of the second second

- Contraction of the second second

(III) - Ha**rana** managana - Hara

Low addr:

High addr:

·····

stradized the state

and the state of the second second second

an a chaile sao in chunn

🖛 🗮 ku terhasi na and see a subsect of the se

a Constanting of the second and the second state of th 🗰 🗶 Arabia (1997) we have a second to the second

and the second

A CALL PROPERTY AND A CALL OF

and the second se

and Constant of Second Malationer of Completion of the

- Arto

entranne consenter St. - Annappendier.

and the second statements of the an an ann an thuis an traja **Carriere**n pop et carrete

網線

and a second Refer to Marine The Marine State of the Stat

0011001101000111	(3,f)
000000000000000000000000000000000000000	(1)
000000000000000000000000000000000000000	(var 1)
0010010001011111	(2, g)
000000000000000000000000000000000000000	(var 1)
100000000000000000000000000000000000000	(anon)
ii	

Figure 3: Skeleton for f(1,X,g(X,Y))

Examples of atom entries and clauses can be seen at the end of the PROLOG sustem code. Even where they have special sustem-defined properties (ie when 3

2

27

Figure

Э.

)

to evaluable predicates) all atoms and clauses must be 2 memory represented as above. If a predicate is associated with a system function, the 9 (and of the clauses can appear there). Then when the head of the clause is matched, instead of looking at the body of the clause the system calls the

2

Ş

special routine.

- 🐅 🗬 en segera de la conse

Million Contractor

and Constanting and the

-1696 S

.

Sec. 10

.....

STAN AN ANTIMA STATISTICS

Weining Addition gelar in program in

the state of the second se

AND COMMENCE OF CONTRACTS

And the second

nie – 🔍 prinkie zwiele – dat obe

Reason and State Services and the services

When a start when a started

in production and the second

AN TRANSPORT

NAMES CONTRACTOR

With Market and the second sec

Mail Mary Late

TRANKENDA STOR Contraction and a state of the Without the second second

An and the second

NUMBER OF STREET

and the second s Shields constant or a set of

stick

滴泡

🗰 🌉 📲 Barressen i Standard van gebru

and the second second second second

NNR STAR COMPANY

the Contemport

an an war i na Carrier and a second

3. Storage of Data in Execution

when the user runs a program, he assigns values to variables corresponding to invocations of clauses. When a clause is invoked, the system assigns a block of storage with 1 word for each variable of the clause. The initial value of each variable is 'undefined', which is represented by the value 0. During the execution, the value may change to one of the following:

3

)

www.www.wwariables are unified. The later instantiation of the second one must result with the same value being associated with the first. The ---- presence of an address where the value of a variable should be says

- An atomic item (inteser or atom). In this case the representation of the item (being the same as in a skeleton) is put in the word.
- A complex term. Complex terms created during execution (DS's) are placed in the slobal stack. If a variable is siven such a value, a pointer to the structure is placed in the variable's word. A DS is see seffectively a copy of a skeleton with values like those we are -currently discussing substituted for the variables that occur in it.

semicles 4. Core Allocation

memory need. First of all, the overall layout of core is illustrated in fisure 4. At any point, the PROLOG system must be able to access the area between the lowest address and GTOP, as well as the area between LTOP and TOP. The former of memory these is in seneral allocated as UNIX data area, with DATLIM siving the address of the highest location that can currently be accessed. The latter is in seneral allocated as UNIX stack area, with STLIM siving the address of the next word below this area. When the stack area attempts to expand over 4K words further expansion is prevented and the data area expands instead to meet it. The flag CONNECTED indicates whether this condition holds. Although the PROLOG system relinquishes control of data space that it does not immediately need (through TRIMCORE), it is unfortunately not possible to do this with stack

Lowest Address:	System code
HASHSTART:	Hash table for atoms
HASHEND:	System clauses and atoms
FREESTART:	User's clauses and atoms (heap)

HTOP>	Free area above heap	3
HLIMIT>	Previous alobal stack areas	
GSTART>	Current soal clause and DS	-
GBOT>	Current global stack	•
GTOP>	Free area between stacks	•
LTOP>	Current local stack	-
LBOT>.	Previous local stack areas	0
TOP>	Top of core	•
n - Einen Neterse gehöttigt der Statistiken - Statistiken	Figure 4: Core Layout	Figur

a 📥 📖 agas 🗤 💡

a de **la sec**aria de la

A second state of the second second

and continues in a second

A CONTRACTOR OF A CONTRACTOR OF

NAME A CONSTRUCTION OF STREET

and the second second

· 時に、 や 高を 維持。 へ 🗰 🗣 Kapal Jakowa 🗉 🕬

With a strength and the

. Ný pro **hou 11 internet in** an agus a cara a sa

authent and calent the rest.

entrinthigge et al a staar

Wing the second second

States - Constant Alline contacts - States - States

THE REPORT OF THE PARTY OF

State of the state

A DESCRIPTION OF THE PARTY OF THE PARTY OF n**er Suisse**en maar aan an an

Wertslad i Meeter

9**6**0

WEAR PROPERTY

195%

Follow Phylinder Sector 1 - 222

.

🙀 📲 tigenford " and there is "

Mit Research and the second second

space. Since it is useful to keep the SP register to handle subroutine calls within the interpreter, SP is not used as a pointer to the stacks at the top of core except when more space needs to be allocated there. In seneral, SP points inside a special workspace area extending downwards from WSTART.

Because of the facility to do nested 'consult's and 'breaks', it is possible that the user may have several different execution environments active at the same time. The main parameters of the current environment are to be found between ENVSTART and ENVEND. The parameters for previous environments are kept on the slobal stack.

Within the space that the system uses, there are basically three main data areas:

-----1. The 'heap'. This is not really a heap, because it is not sarbase collected. Instead it is a stack containing all the permanent clauses and atom dictionary entries. This area starts at FREESTART a wandwaits top is siven by HTOP. The expansion is limited to below -----HLIMIT, which is where the slobal stacks start. If expansion further is attempted, the slobal stacks are automatically moved up to make -----space. Anything placed on the heap must start with a word with an even value (because there are no explicit clause terminators). Note that the function to check the availability of space on the heap (HALLOC) does not update HTOP. This is so that it is possible to put values there but to postpone making them permanent until the last 5

minute. In this way one can avoid permanently messing up the heap if an error occurs half way through putting something on.

2. The slobal stack areas. There is one slobal stack for each active user execution. The first one's start is given by HLIMIT and the current one's by GBOT. The top of the current one is given by GTOP. The alobal stack is used to store terms constructed during the ---execution (referred to in many comments as DS's). In addition, the DS where GBOT points, pointed to by GSTART. A second function of the A mention of the stack is to contain the trail - the list of variables whose values must be reset on backtracking. Trail entries are interspersed with DS's on the global stack; they can be told apart because a trail entry is a single address (even value) whereas a DS starts with an odd value. If the stack is moved up, the pointers GBOT, GSTART, OPSTART, HLIMIT and GTOP are updated, as are any pointers into the slobal stack from itself or the local stack. No other pointers are updated. Futting anything on the global stack must be preceded by an appropriate call of GALLOC.

> 3. The local stack areas. There is one local stack for each active user execution. The local stacks expand downwards, whereas the slobal stacks and heaps expand upwards. The first local stack begins at the address siven in TOP. The current one besins at the value of LBOT and extends to the value of LTOP. Local stacks are used to contain administrative information about the user's execution as well as cells for variable values. These are allocated in groups called stack frames. Putting anything on the local stack must be preceded by an appropriate call of LALLOC. Note that the part of the stack used during the satisfaction of a goal will be reclaimed if this succeeds determinately.

5. Keeping track of Execution

%液:

1000 -Marine .

The Constant American Constant

· ·

1. Har Mar Martin

all a construction of the 10-200 Aug

ALIMENTER AND COMPANY

A CONTRACTOR OF CONTRACT OF CONTRACT

and the second second

W ALWARDS

and the second second second second

and the second sec where the state of the state of the state

ngga 🗣 jaggarik (mina) - - - - -

n **an** Alberta (Carl The Carl

HIGH WEITER

and the second second at a second a

Anter Statistics of the second

and the second second

and the second se

unter anti di Gardi Maria di Cara di C WARE AND A PROPERTY OF A CONTRACT OF A CONTRACT

um Sussianos e com com com

and the second second

Meratana ang Ang Para

an the first sector and the

Caller of the second

a**n C**hanning and

· 14 · ·

1 Barrow Barrow Barrow

Salaria Salatina da ...

a**n an an a**n tha an t

Silen Restation

1403.0

्राक्षील

willing .

i san

-

....

The code for the main loop of the interpreter (from CONTINUE to GOON) is fairly well commented, but the main structure can be summarised here.

Given the skeleton for a goal to be satisfied (at GOTCALL) the system obtains the atom which acts as its functor and finds the first clause with that atom as predicate (now at GOTCLAUSE). It must then allocate a stack frame on the local stack for a new invocation of the clause. It starts off by recording administrative information and then (at ENTER) allocates space for the variables of the clause. It then invokes the unification routine (UNIFY). One argument to this is a pointer to the skeleton that mentioned this goal, together with a pointer to the stack frame which gives the values of the -variables in the skeleton. The other argument is a pointer to the skeleton at the head of the clause, together with the address of the new stack frame (siving the values of the variables for the new invocation).

6

Q1,

e.:

an more than the second of the second s • second a system clause has been activated. If not, the search for the next soal can - take place immediately. The first place to look is after the head of the new was well and there is a soal there, that is fine; otherwise we have completed the current soal. In this case (now at RETURN) we must look back me Character and so an until a remaining unsatisfied soal can be found. If this process soes back right to the original user soal, we have been successful (SUCCESS).

momentum relates for the predicate, backtracking takes place (FAIL). After resetting the values of the variables given in the e strail, the system looks for the last choice point (The address of its stack) momentary of the set o

٩. (Now back to GOTCLAUSE).

. The soal that the user (at EXECUTE)? The soal that the user www.sec.esives is 'asserted' as the body of a clause for the predicate '' (the atom with emmanate an empty name). That is, the atom entry for '' (at EDUMMY) is made to point to a dummy clause at the start of the slobal stack. The system then sets the address (DCALL) of a dummy soal of '' and starts the interpreter cycle at GOTCALL.

١.

£.

The details of the interpreter loop can only really be understood if the format of stack frames is explained. Figure 5 does this.

6. Unification

n a the second of the second second

咽髓炎

(時間)

transferrenden an of an and an

nd **i dat** best sid

New States and States and States

CONTRACTOR AND A SALE OF A SALE

AND THE PROPERTY AND THE P en definisien.

NITREASE STATISTICS

a sector and the sector of the

AN TO BE AN ADDRESS OF A --

And Minister and the

and the subscription of th Statistics for statistical with the state of the state CONTRACTOR PAGE COMPRESSION the state of the second second C. Networks William State of the Contempt

www. A CONTRACTOR OF THE OWNER OF THE References -And the second second second second

2844 100

(interim)

- During the execution of a user's program, there are two ways in which ٩. memory instantiations of the skeletons in his clauses can be characterised. Firstly, memory siving the address of a skeleton plus the address of a stack frame (normally the "base" address) sives enough information to specify the term denoted, Because the skeleton sives the basic structure and the stack frame sives the manager values of the variables that fit into that structure. Secondly, giving the 🐜 address of a concrete DS is also a complete specification. The basic menorementation of the similarities in the representation of skeletons and DS's and work for both types of representation. The only ma happens in a skeleton) the appropriate environment (stack frame) must be wave available, so that the value of the variable can be looked up. The locations magnetic set with the set of the

where the second s term and also to return an appropriate value result. It can be applied to a ********** ---- variable cell or to part of a skeleton or DS. If it is used on a skeleton, the present appropriately. The possible results that IDENT can return are given in fig 6. Most of what IDENT does is just following

7

 Statistics of the probability of the statistic statistics. 		and the state of the
s makeow addr to see a	Cell for var n	en la fil
and the second		n - Carron Weiter gr
್ರತ್ ಎಂದ್ ವರ್ಷ (ಕ್ಷಣ್ಮ ಕಾರ್ಟ್ ಪ್ರಕರ್ ಕಾರ್ಟ್ ಪ್ರಕರ್ ಕಾರ್ಟ್ ಕಾರ್ಟ್ ಕಾರ್ಟ್ ಕಾರ್ಟ್ ಕಾರ್ಟ್ ಕಾರ್ಟ್ ಕಾರ್ಟ್ ಕಾರ್ಟ್ ಕಾರ	11 m · m · m · m · m · m	e e envita
$= \left[\partial_{\mu} + \partial_{\mu} \partial_{\mu} + - \partial_{\mu} (1 + e_{\mu}^{-1} + \frac{1}{2} + 2e^{2\pi i \frac{1}{2}} + 2$		n a second
$e^{-i\phi}(r_{i})^{1}$, suggestive theory of the set of		to a second seco
an t he same of the assess of the state of the second second second second second second second second second se	Cell for var 1	
erenden van die entwikken opplikken en wetten ander die en die		$1 \sim 12^{\rm eff}_{\rm He}$
na ana mangaletta mengela atau da mandar da m	Addr of clause	and the second sec
a sandar in suit that the state of the transmission of the second	being used	E E - Constantino E
al and the providence of the Second Constant of the second		s de la construcción de la constru Esta de la construcción de la const
n - Gyna th - Maetan Alger - Angalan Inger - Angel - Ang	Addr of call of	
1. 19. Million - Martin Barting and Martin Statistical Condition of the	this soal	t
$\label{eq:started} \mathbf{x}(t) = \mathbf{x}(t) \mathbf{x}(t) + \mathbf{y}(t) \mathbf{x}(t) \mathbf{x}(t) + \mathbf{x}(t) \mathbf{x}(t) \mathbf{x}(t) + \mathbf{x}(t) \mathbf{x}(t) \mathbf{x}(t) \mathbf{x}(t) + \mathbf{x}(t) \mathbf{x}(t) \mathbf{x}(t) \mathbf{x}(t) + \mathbf{x}(t) \mathbf{x}($		1
and the second	Base of frame for	{

	I parent gan]	Le constante de	Č , 1
n anna an Na Martine - Sairte Anna Anna Sairte Sairte - Sairte Anna Sairte - Sairte - Sairte - Sairte - Sairte - Sairte -			
na an ann an an ann an ann an air an an an an ann ann ann ann ann ann an	Base of frame for last choice point	<pre>> Only > present > case</pre>	se se
An Antonio and a second s	<pre>Value of GTOF When frame created </pre>	I / Por I / Choice II / Points	station Station
na na sana na s			
Mennest Processes of the second second	Figure 5: Stack	: Frame Representation	Fische
Pro Michighton (Methoda La Scott)			
n an the state of the second state of the seco	Type result Valu	e result	-
ANY AND THE AND THE A MERICAN A CONTRACT OF A STREET AND A	(in RO) (in	R1)	,
- tm			

0

2

4

6

8

.

variables in environments. The main unification routine UNIFY is normally called with two skeletons and environment pointers. These correspond to the call of the current soal in another clause and the head of the new clause that is being tried. In fact it will work with any two things that IDENT can interpret. The process of unification involves following the tree structure of the two terms, checking that functors are equal, and then doing special actions whenever a leaf of a tree is encountered. If UNIFY performs a normal subroutine return, unification has been successful and all the necessary substitutions have taken place. The only other possible action is for a jump to FAIL to take place.

Figure 6: Possible Results of IDENT

The representation of an integer

The representation of an atom

The address of an undefined variable

The address of a complex skeleton/DS

The location of an anonymous variable

Figur

€.

UNIFY starts by calling IDENT for each argument, and then jumps to an appropriate routine according to the combination of types. Some routines just 8

involve comparing two values (TESTEQ), some mean immediate failure (UNIFAIL), and some involve simply assigning a value to a variable (VTO1, VTO2, ASS2V). If one variable is made to point to another, it must be ensured that pointers so in such a direction that if pieces of stack are reclaimed there will be no pointers into limbo. The rules for ASS2V make sure of this (In this context, a 'local' variable cell'is a cell in the local stack whereas a 'slobal' cell is one that appears as part of a DS on the slobal stack). Moreover, whenever any value is assigned to a variable, the address of that variable must be put on the trail if the value has to be reset on backtracking (Routine TRAIL).

The more difficult cases occur when complex terms are involved. If two complex terms are being unified (SA12) a certain amount of manoeuvring is necessary to take account of whether the terms are part of larger terms in prefix format or not. Then unification is called recursively on corresponding arguments (assuming identity of functors). If a complex term is unified with an uninstantiated variable, two possible cases arise, according to whether the term is in the form of a skeleton (when it appears below GBOT) or a DS. If it is a DS, it is a simple matter to put its address in the variable cell. Otherwise a copy of the skeleton is put on the global stack (routine FLESH), with appropriate values substituted for the variables. (This is where DS's are created). The variable can then point to this.

7. Routines for Evaluable Predicates

with the second second

and the second second

togiðlar still stillagaðinnar

Ale water and the

de logge sal tode - - n 🖗 💭 de la comercia

Hereine Parante

Myressi who

林静主义者 建化化

nam i kan natiti ka in

an Same

and a second second

· Partition of the analysis of the second second

and the state of the second state of the secon

🚛 🧶 . There is a star of a same of a 👬 👘 🕂 Series de la constant de la

AND FINE OFFICE STREET, ST

With the second s

San Santa

THE PART OF A CONTRACT OF A

And the second second

9

stiller - ----

. 0 $+ \Theta ((\mathbb{R}^{n}))^{n} (\mathbb{R}^{n})^$

KARAGASIAN ALIJAR COMPANY STATE

Sig. a

AND THE COMPANY

()出来 ()

e contra con

The way in which special routines are invoked for evaluable predicates was discussed in earlier sections. This section gives some guidelines for writing such routines.

When a routine for an evaluable predicate is called, R1 points to the 'variable 1' slot of the current stack frame. The variable ARGPTR also holds this address. The normal procedure is to start by checking the types of the arguments, through IDENT, storing the values as necessary (All the registers except SP are available in these routines). The the work for the predicate is done, and a normal subroutine exit indicates successful satisfaction of the soal.

If the soal is not satisfiable, it suffices for the routine to cause a jump to FAIL. FAIL will clear up various bits and pieces, and does not require any

registers to be set. Alternatively, if a serious error occurs, the routine un means the second second be called. This must be a proper subroutine call, even though the call will never return. The number of the error should be placed in the word after the subroutine call, so that it can appear in error messages.

ments are the second points to note when writing evaluable predicate routines are the following. Firstly, if a value is assigned to any variable cell, TRAIL must be called. Secondly, any additions to the stacks or heap should be preceded by calls of the appropriate ALLOC functions and must take into account how items manage may have shifted by the time it returns - so if one has pointers into the

. 9

🐜 🚛 🐜 alobal stack in such situations it is necessary to use OPSTART or keep addresses relative to some pointer that will be updated (es GBOT). Finally, the fact that the user can specify 'abort' to happen immediately after an interrupt means that evaluable predicate routines must be prepared to be discontinued at any point. This means that permanent additions to the heap or changes to system variables must be carefully timed or interrupts disallowed for their duration (through INTROFF and INTRON).

Finally there is the task of producing dummy clauses and atom entries for new evaluable predicates, and in particular of obtaining the atom number of a new predicate. Unfortunately there is no way of doing this properly (yet). The best way to discover what the atom number of an atom is seems to be to run PROLOG, type in a simple clause with the atom as predicate and then look at the core magneeing of the second se FREESTART.