Accessing Calendars and Schedules

This subsection covers the following topics:

- Accessing Dates within Calendars or Schedules
- Example for Using NOPUCS1N

Accessing Dates within Calendars or Schedules

You can access dates within calendars or schedules using the following statement in your Natural application:

```
CALLNAT 'NOPUCS1N'
    P-FUNCTION P-RC P-OBJECT-TYPE P-OWNER P-OBJECT P-DATE-A8
```

Meaning of the parameters:

Parameter	Format	Usage		
P-FUNCTION	A01	in	Function code:	
			C	Determine the calendar used.
			D	Determine the schedule used.
			N	Determine the next set date, from a starting date.
			P	Determine the previous set date, from a starting date.
			R	Reset a date. Date will be removed from calendar or schedule.
			S	Set a date. Date will be added to calendar or schedule.
			T	Test a date. If date exists within calendar or schedule: **RC** contains 0, otherwise 1.
P-RC	N03	out	Return code:	
			0	Function ok; or: date is set.
			1	Date is not set.
			2	Object not found.
			101	Invalid function code.
			102	Parameter(s) missing.
			103	Invalid parameter combination.
			104	Invalid object type.
			105	Invalid date.
			106	Table of explicit dates full.
P-OBJECT-TYPE	A01	in	**Object type:**	
			C	Calendar
			H	Schedule history
			S	Schedule
P-OWNER	A10	in	Owner of object.	
P-OBJECT	A10	in	Name of object: calendar name, schedule name (=network name).	
		aus	Funktion codes **C** and **D**: name of determined object.	
P-DATE-A8	A08	in	Date in format: YYYYMMDD	
		aus	Function codes **N** and **P**: the date found.	

Notes:

1. Only function **T** is allowed for object type **H** (schedule history)
2. DATE-D should be used for calls from Natural programs.
   P-DATE-A8 must be empty in this case.
3. Schedule history is usually available for the current year and previous two years.
4. Schedule dates can be set for the current year and the next year.
5. The functions **N** and **P** work for a range of years.
   Limitations:
   - For schedules, only the current and the previous year are available.
   - Calendars must be defined for all years in which a search is to be performed.

 Copyright Software AG 2003

Example for Using NOPUCS1N

The program calculates the first Monday of each month in 2001 and subsequently enters the network REQUEST/BFA001 for the calculated days in the schedule.

```
* BFA001
* DETECT FIRST MONDAY IN MONTHS
*
DEFINE DATA
LOCAL
1 #YYYYMMDD (A8)
1 REDEFINE #YYYYMMDD
 2 #YYYY    (N4)
 2 #MM      (N2)
 2 #DD      (N2)
1 #D1      (D)
1 #WD      (A1)
1 #OWNER   (A10) INIT <'REQUEST'> /* Owner   < ---------------------
1 #OBJECT  (A10) INIT <'BFA001'>  /* Network < ---------------------
1 #RC3     (N3)
END-DEFINE
*
#YYYY := 2001                     /* Year    < ---------------------
F1. FOR #MM = 01 TO 12
 RESET #DD
 R1. REPEAT
  ADD 1 TO #DD
  MOVE EDITED #YYYYMMDD TO #D1 (EM=YYYYMMDD)
  MOVE EDITED #D1 (EM=O) TO #WD
  WRITE #D1 (EM=YYYYMMDD) #WD
  UNTIL #WD = '1'      /* R1.
 END-REPEAT        /* R1.
 WRITE #D1 (EM=YYYYMMDD' 'N(10))
*
* CALL NOP SCHEDULE API TO SET THESE DAYS
*
 CALLNAT 'NOPUCS1N'
  'S' #RC3 'S' #OWNER #OBJECT #YYYYMMDD
 WRITE #YYYYMMDD #RC3
END-FOR      /* F1.
END TRANSACTION
END
```