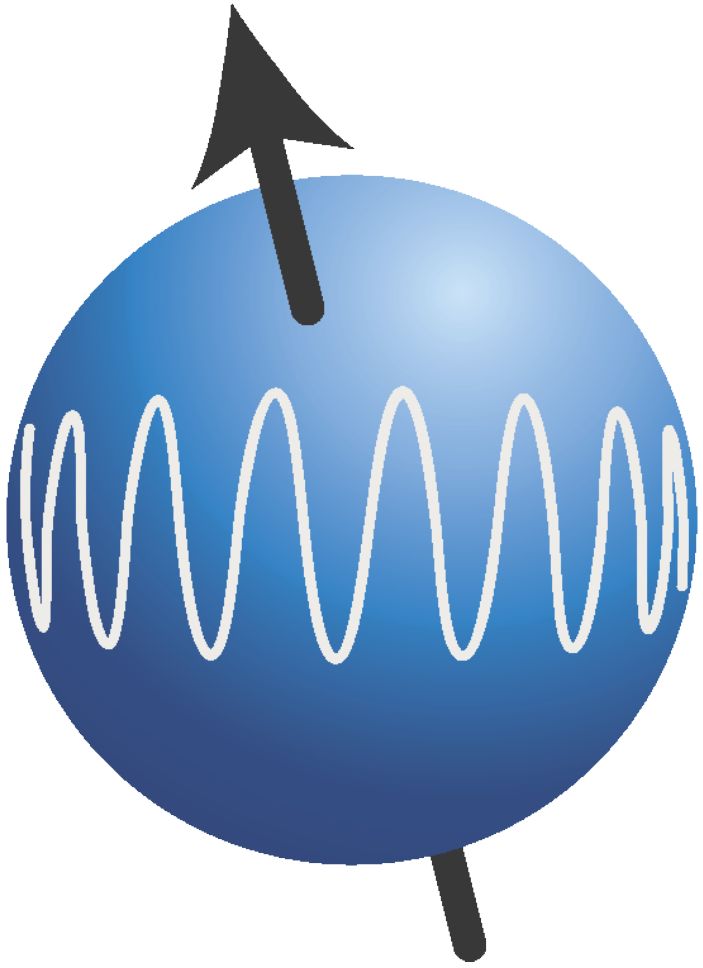


TopSpin

Method Development

Contents

→ Pulse Programs

→ Shapes

→ Experiment Development & Simulation (NmrSim)

→ AU Programs

→ Python Programs

→ Macros

→ Pulse Programs In Python

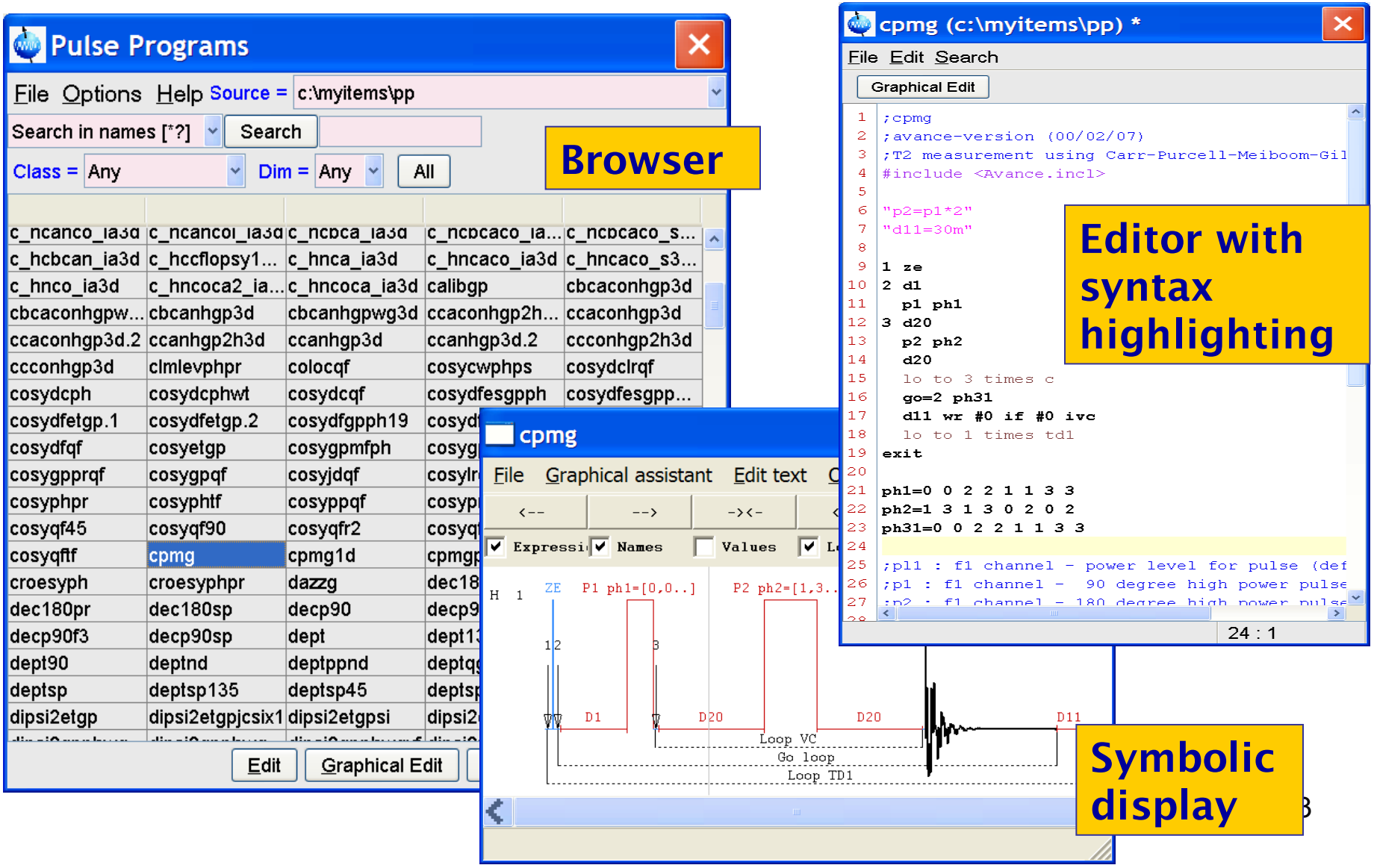
→ Plot Layout Programming

→ User-Defined Command Panels

→ *Please click on a topic to view page*

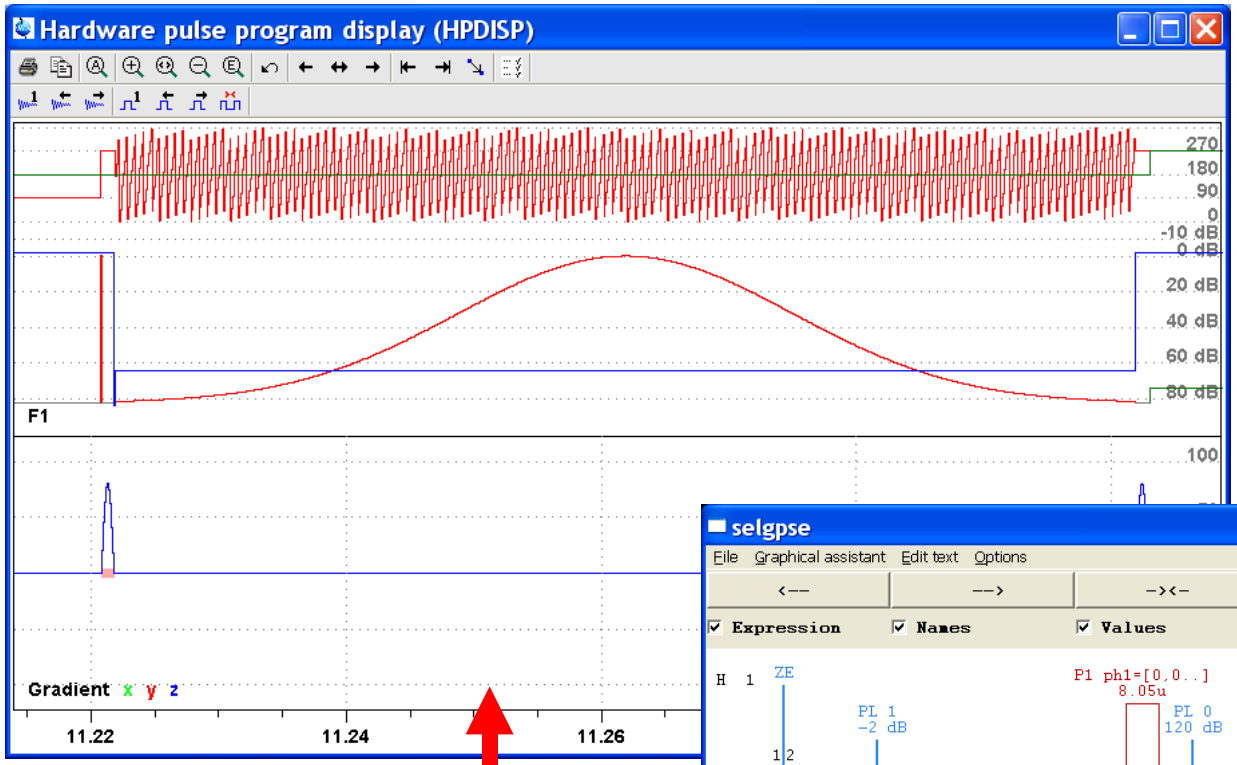
Pulse Programs

Development Tools



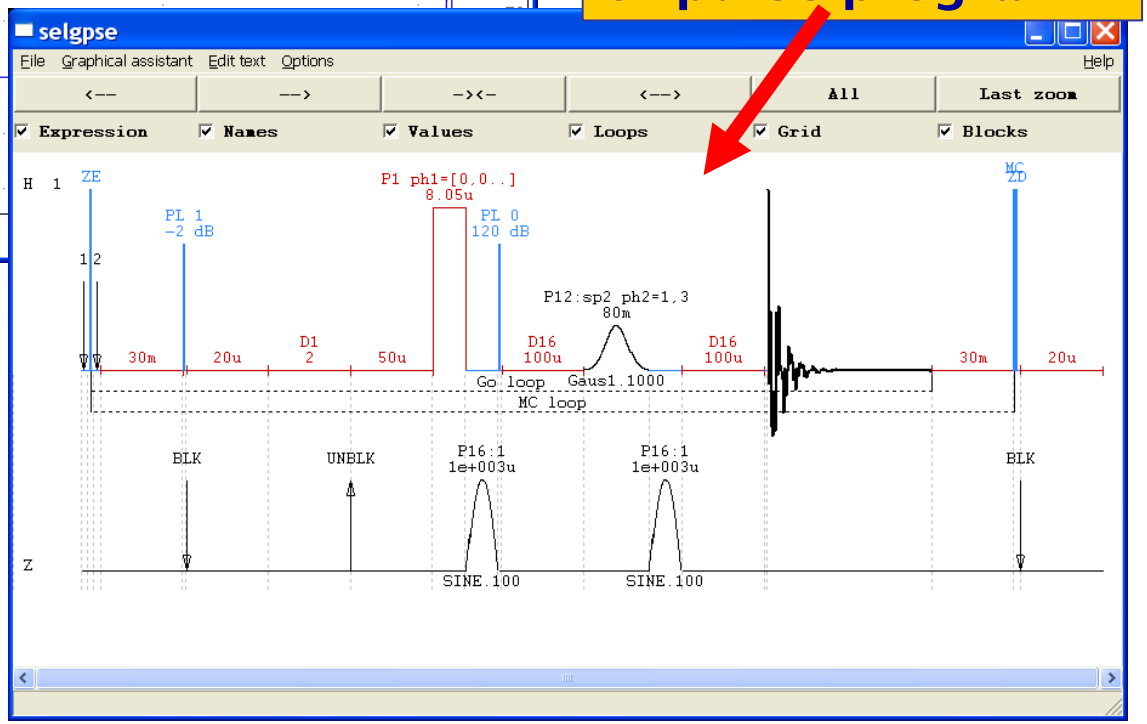
Pulse Programs

```
1  ze
2  30m
   20u pl1:f1 BLKGRAD
   d1
   50u UNBLKGRAD
   p1 ph1
   p16:gp1
   d16 pl0:f1
   p12:sp2:f1 ph2:r
   p16:gp1
   d16
   go=2 ph31
   30m mc #0 to 2 F0(zd)
   20u BLKGRAD
exit
```



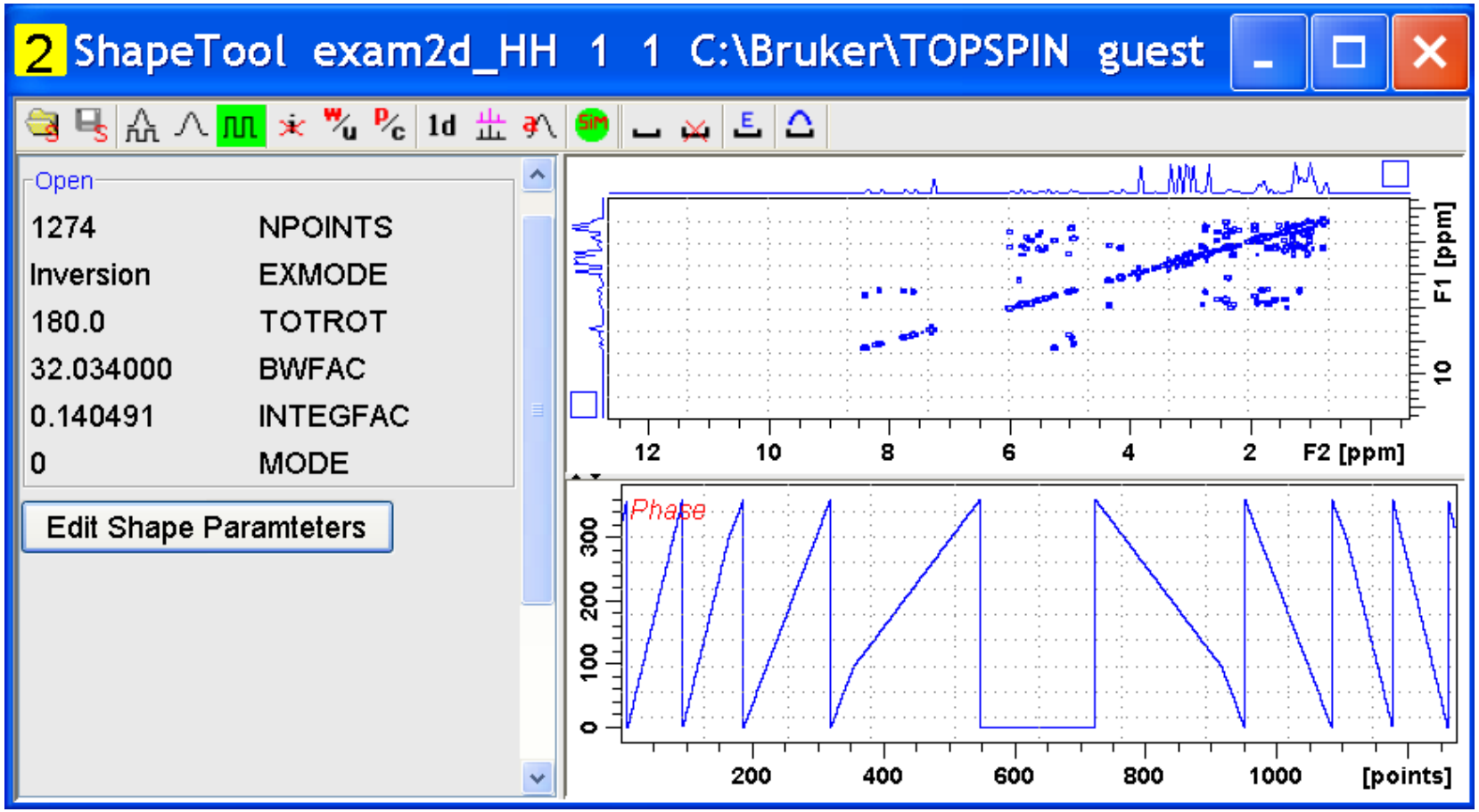
Simulation of pulse generator output for pulse program development & testing

Symbolic display of pulse program



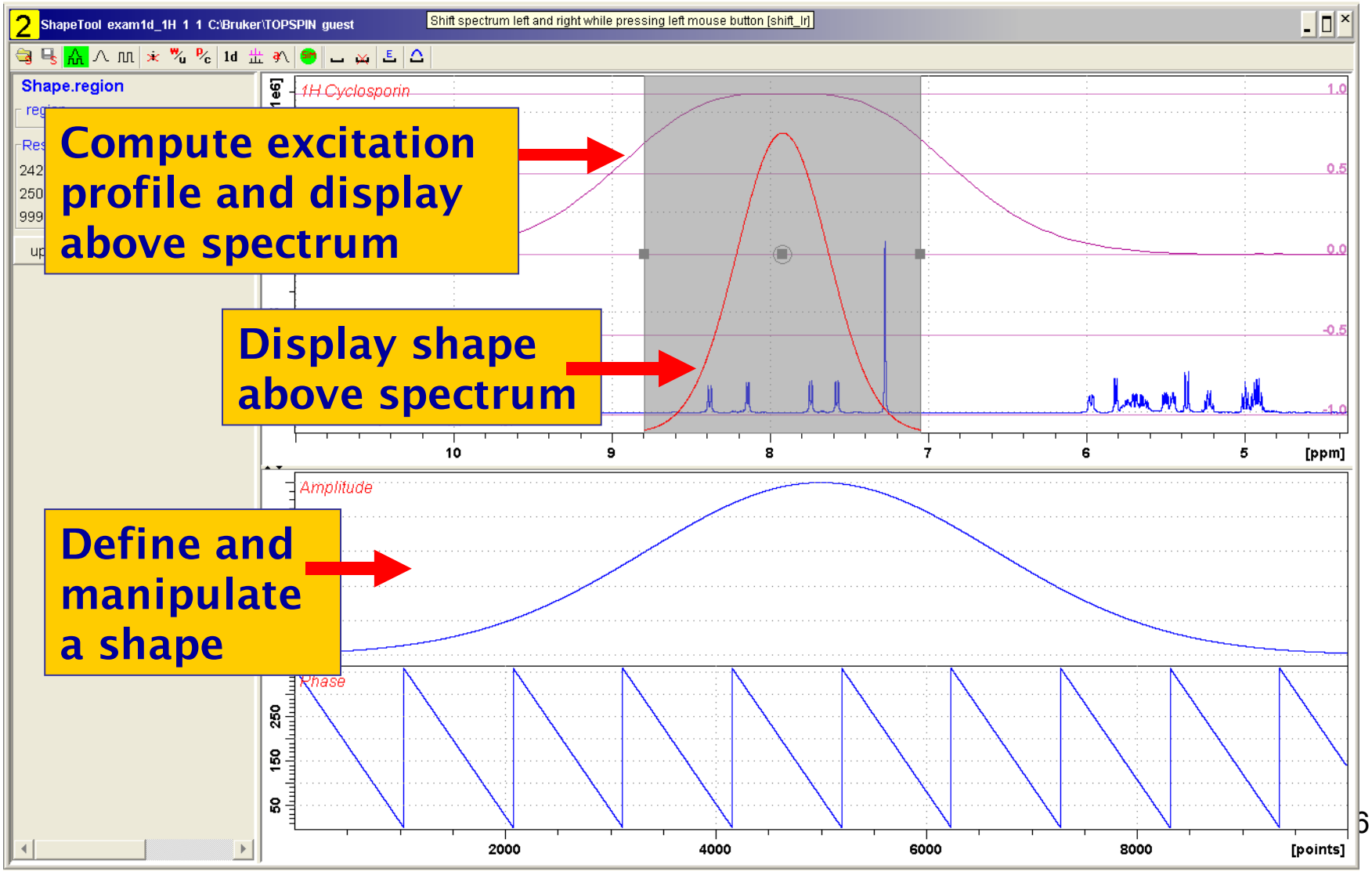
Shapes

ShapeTool to define shaped pulses



Shapes

Calculation and display of excitation profiles



Experiment Simulation (NmrSim)

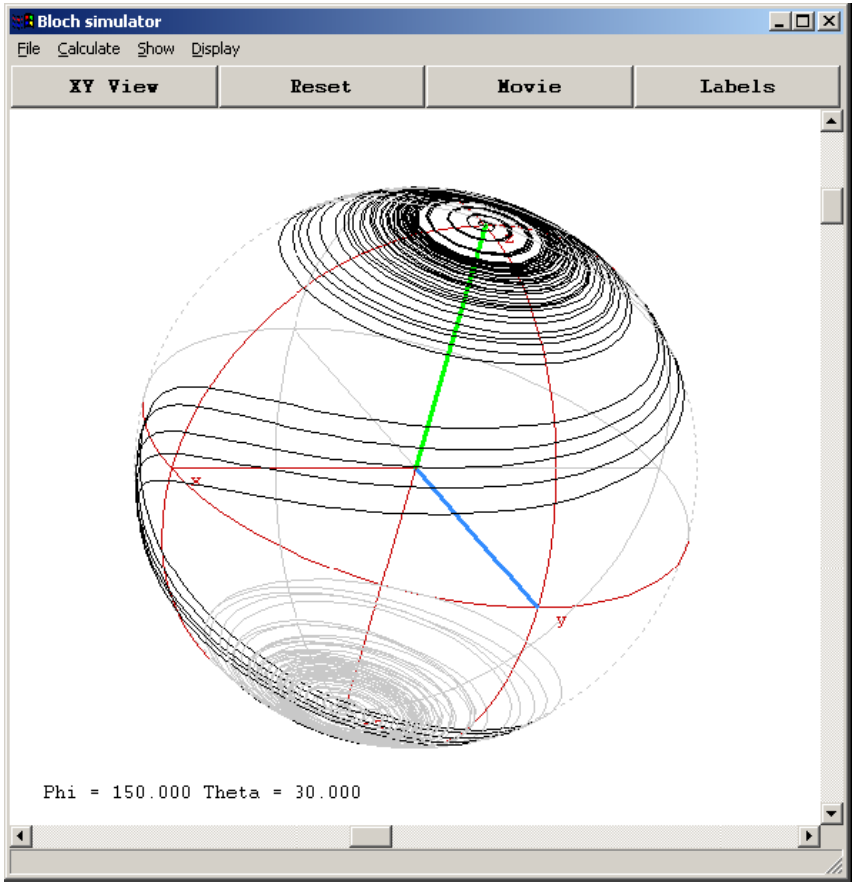
BRUKER

- Simulates a wide range of NMR experiments
- Simulation result is a 1D, 2D, … fid
- Simulates the behaviour of *general homo- and heteronuclear* spin systems (up to quantum number 2)
- Supports simulation of rotating frame magnetization transfer experiments (TOCSY or HOHAHA)
- Supports simulation gradient enhanced exp.
- Solves the quantum mechanical Liouville equation

Experiment Simulation (NmrSim)

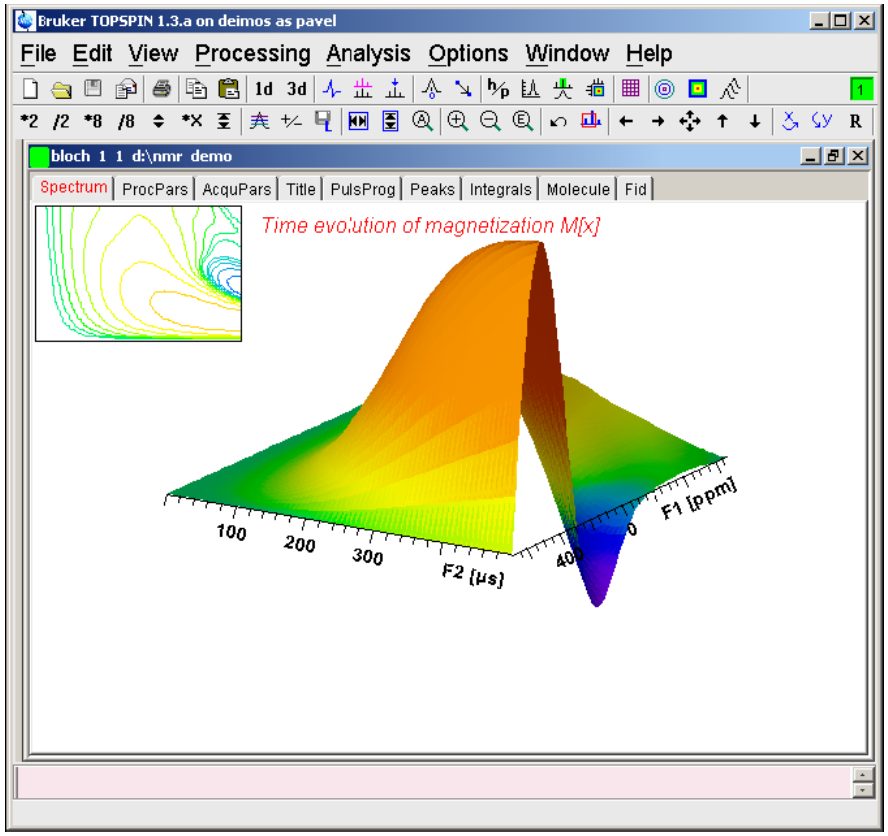
The Bloch module lets you visualize the time development of the nuclear magnetization during various experiments.

The time evolution of a nuclear magnetization vector during an adiabatic pulse, calculated for several offsets, is shown as a projection on the sphere



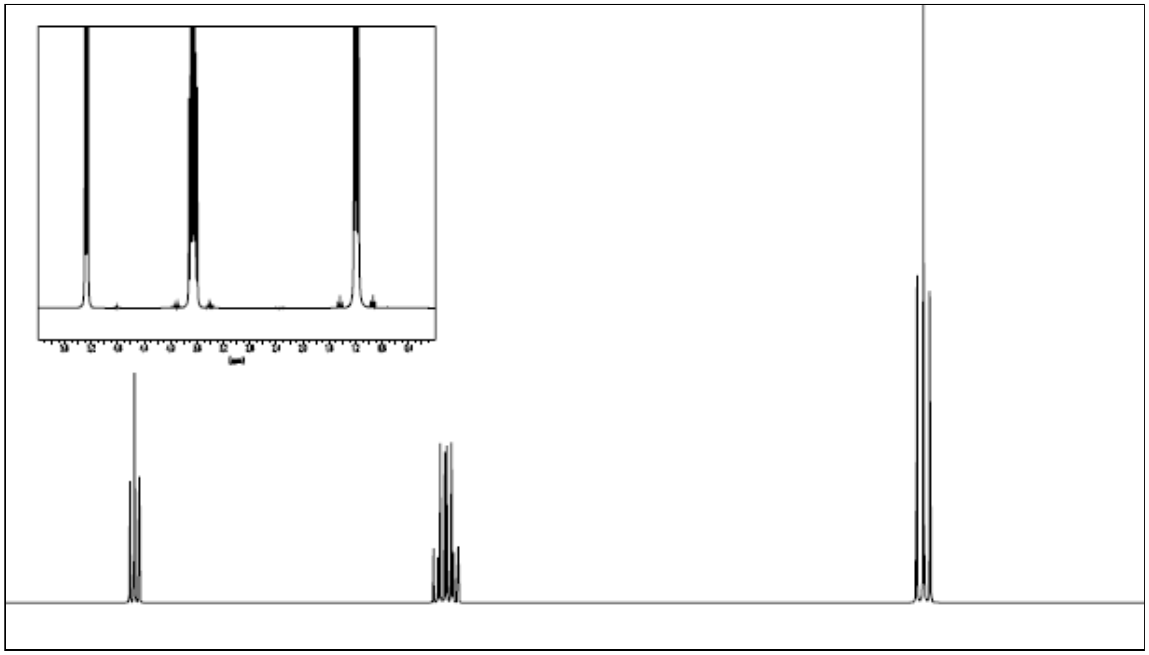
Experiment Simulation (NmrSim)

Time evolution of the magnetization shown in TOPSPIN



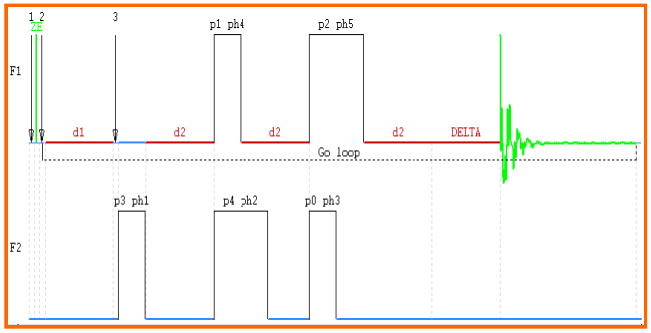
Experiment Simulation (NmrSim)

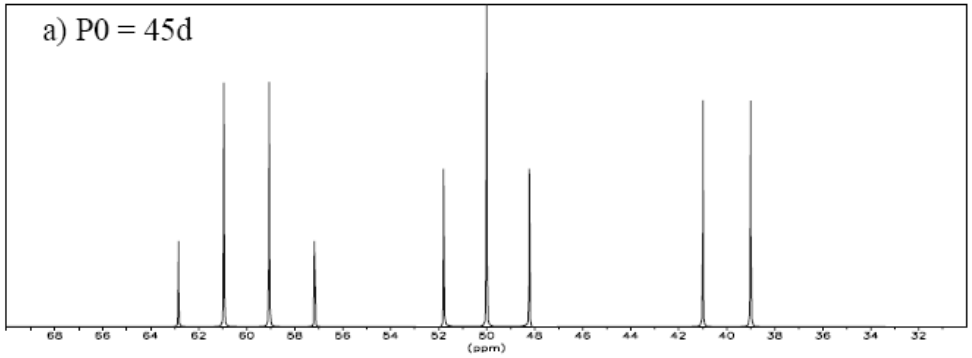
The proton spectrum of ethanol at 200 MHz. The inset shows the C13 satellites (vertical scale with 8 times magnification).

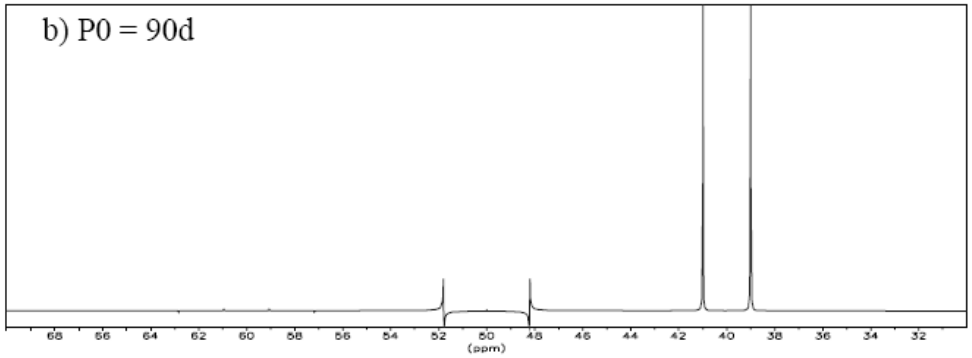


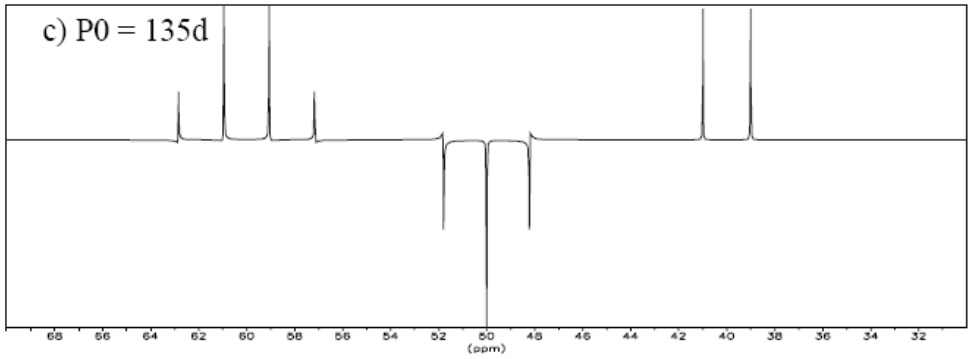
Experiment Simulation (NmrSim)

Comparison of simulated 13C DEPT spectra for three different leading pulses P0.



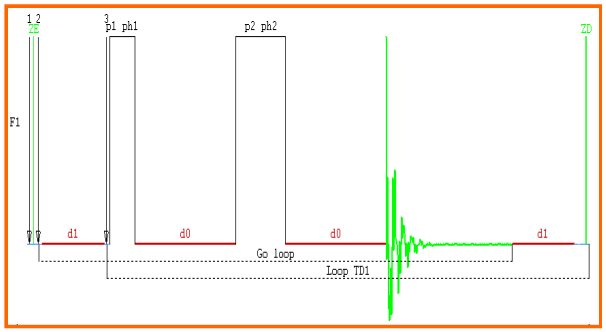




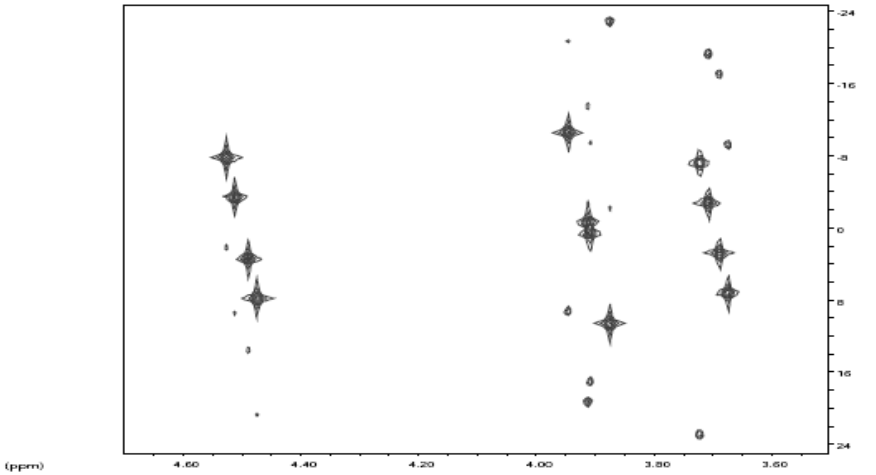


Experiment Simulation (NmrSim)

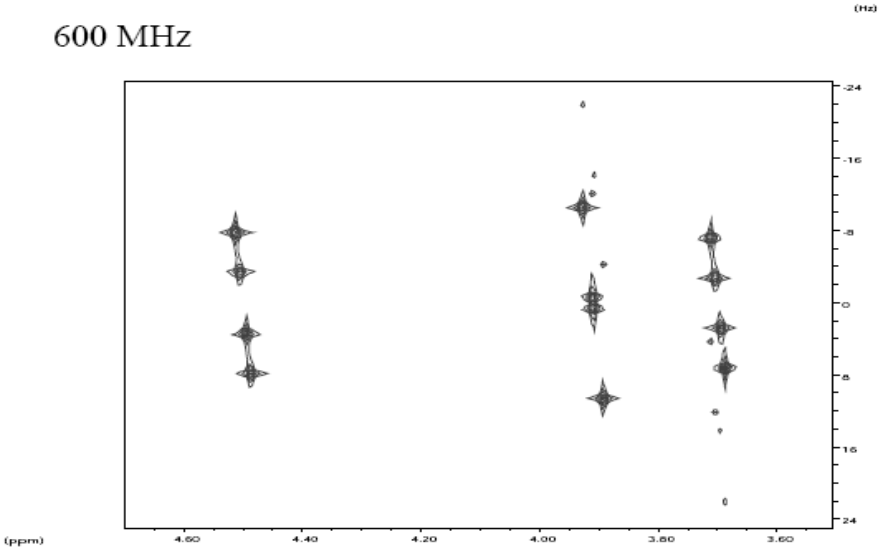
Comparison of J-resolved spectra on 300 MHz and 600 MHz shows the significant reduction of *strong coupling artifacts* for the higher resonance frequency



300 MHz



600 MHz



AU Programs

What is an AU (Automation) program?

A C program automatically interfaced to TopSpin

Purpose:

Extending TopSpin with user-defined functionalities, programmed in C

Example:

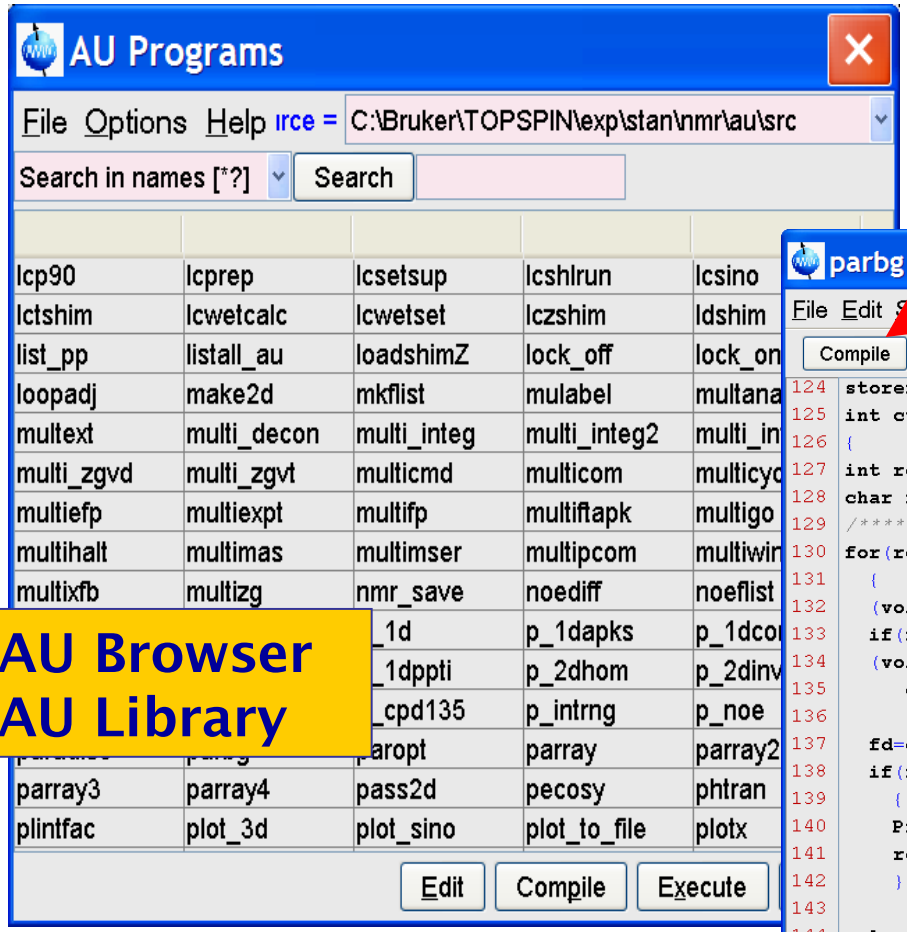
Requests LB, performs em,ft,pk on the current dataset

```
double lb = 0.3;

GETCURDATA
GETDOUBLE ("Enter LB: ", lb)
STOREPAR("LB", lb)
EM
FT
PK
QUIT
```

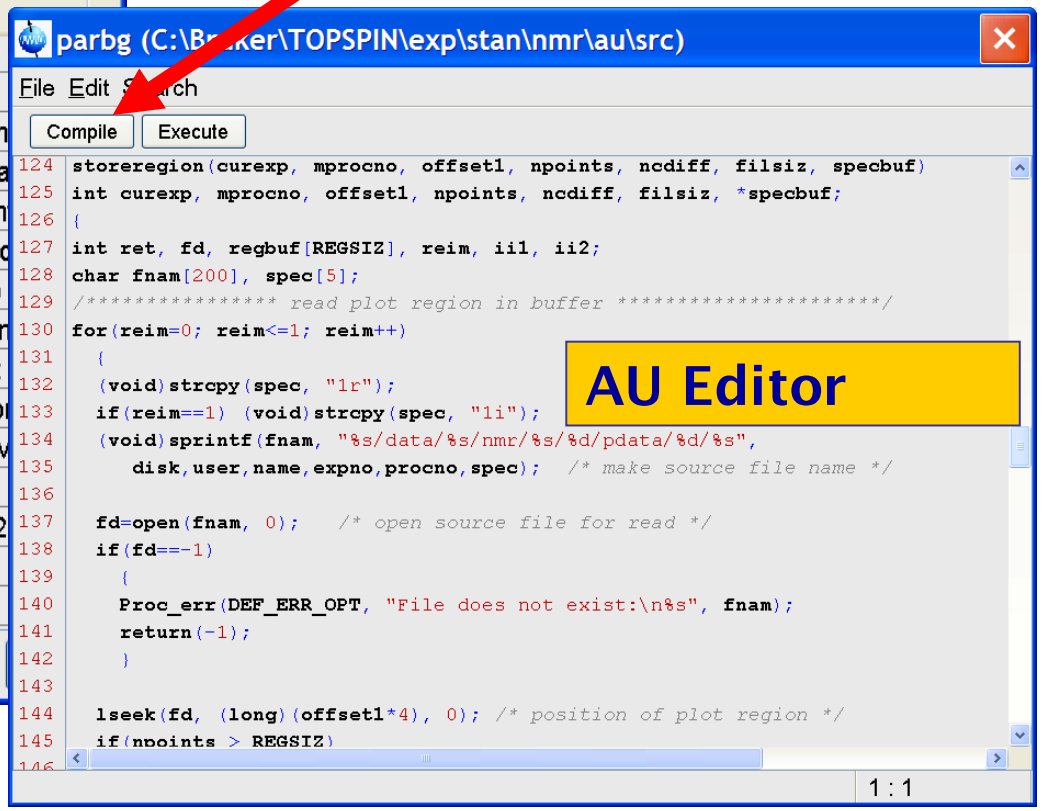
AU Programs

AU development tools delivered with TopSpin



AU Browser
AU Library

GNU C Compiler
For Windows or Linux



AU Editor

Python Programs

TopSpin supports: ***Python programs automatically interfaced to TopSpin***

Purpose:

Extending TopSpin with user-defined functionalities, programmed in Python ("Jython")

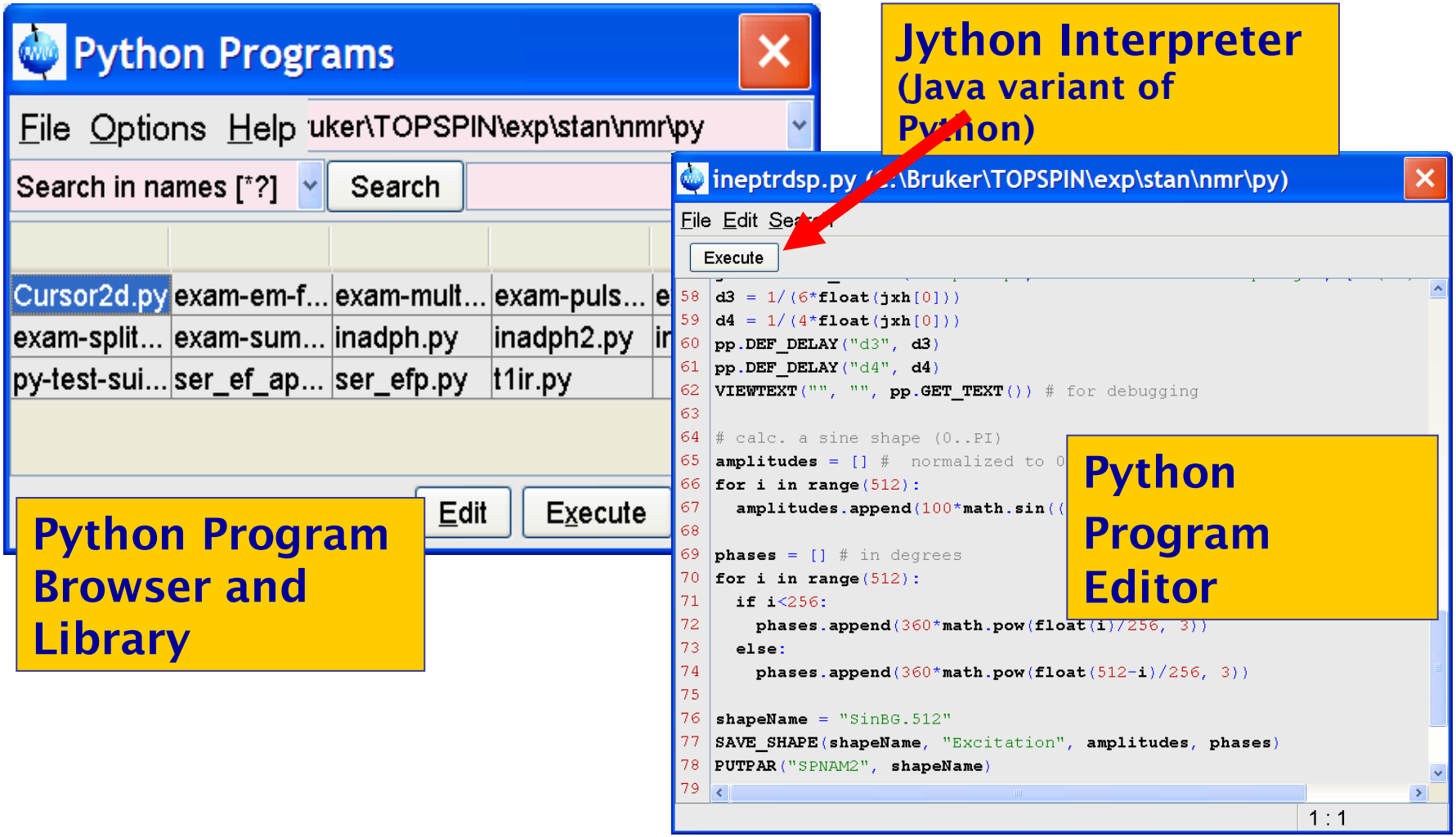
Example:

Requests LB, performs em,ft,pk on the current dataset

```
lb = INPUT_DIALOG("", "",\
    ["Please enter LB: "], ["0.3"])
if lb == None:
  EXIT()
PUTPAR("LB", lb[0])
EM()
FT()
PK()
```

Python Programs

Python development tools delivered with TopSpin



Macros

What is a Macro?

A sequence of TopSpin commands

Purpose:

Extending TopSpin with user-defined functionalities, built from existing commands

Example:

Sets LB, performs em,ft,pk on the current dataset

```
LB 0.3
em
ft
pk
```

Pulse Programs in Python

BRUKER

What is a Python Pulse Program?

A Python program acting as a Pulse Program

Purpose:

Computing pulse program parameters such as delays, shapes, phases using Python statements for the contained pulse program

Example:

Computes phase list ph1 using Python and starts acquisition

```
PPTEXT = """
1 ze
2 d1
  p1 ph1
  go=2 ph31
exit
ph31=0 2 2 0 1 3 3 1
"""
pulsprog=DEF_PULSPROG(PPTEXT)
ph1 = [0,0,4,4]*2 + [2,2,6,6]*2
phlinc = 360/4
for i in range(16):
    ph1[i] = float(ph1[i])*phlinc
pulsprog.DEF_PHASE_LIST(1, ph1, phlinc)
PUTPAR("PULPROG", "bgzg")
pulsprog.SAVE_AS("bgzg")
ZG()
```

User-Defined Command Panels

What is a command panel?

A panel containing user-defined buttons to execute commands, AU programs, macros, etc.

Purpose:

Defining application specific user interfaces

Examples:

script *result*

```
BLUE1=51$ 204$ 255
YELLOW1=255$ 255$ 0
GREEN1=84$ 196$ 20
# Title definition
TITLE=1D Processing Panel
TITLE_COLOR=0$ 0$ 255
# Toggle button definition
TOGGLE_BUTTON=To 2D
TOGGLE_CMD=bpan bproc2d
TOGGLE_TIP=Switch to 2D processing
# Top row button definition
TOP_BUTTONS=EM$ $FT$ $PK$ $
TOP_COLORS=YELLOW1$ YELLOW1$ YELLOW1
TOP_CMDS=em$ ft$ pk
TOP_TIPS=Exponential multiplication $\
Fourier transform$\
Phase correction
```