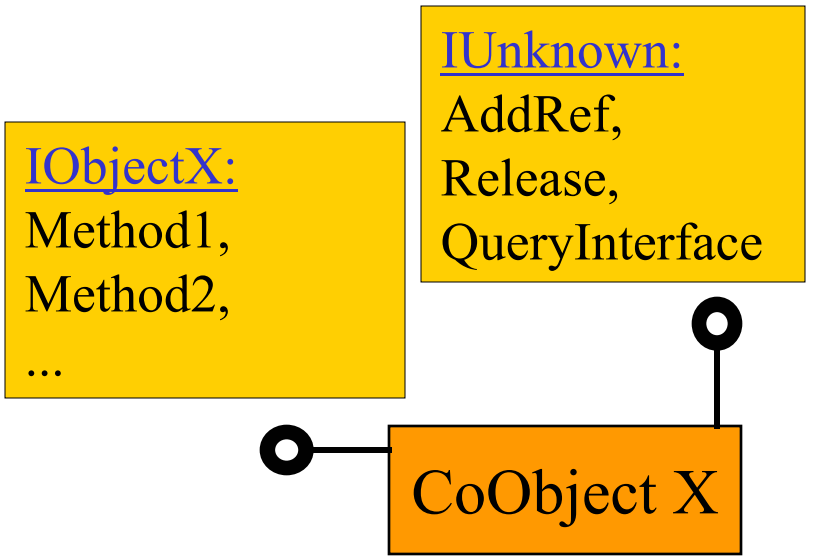
EPICS OPC Gateway

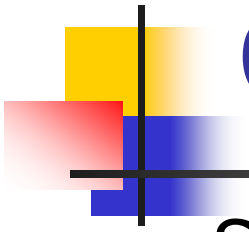
Nov. 2000 ... May 2002

CA Client->OPC:	Andrei Makhnachev,DESY
OPC-> CA Server:	Kay-Uwe Kasemir, LANL
OPC Experience:	Matthias Clausen, DESY
Investigating:	Sverdrup (TN, SNS conv. Fac.)

OPC?

- OPC: ~~OLE~~ COM for Process Control
- COM: Component Object Model, Microsoft/Win32 API
  - CORBA w/ actual application support
- All COM Objects have Interface *IUnknown* plus interfaces to implement an...
  - ActiveX Component/Widget
  - ActiveX Automation Server: (Visual Basic, Interpreters)
    - OPC Data Access Server/Client
  - ....





OPC Data Access Details

- Specs: http://www.OPCFoundation.org
- Data Access Object Model:
  OPC Server, Group, Item
- Other Interfaces:
  - Browsing
    (Info for humans, does not show what items can be created dynamically)
  - Subscribe to group (max. rate, deadband) for async. notification
  - Read/write item data (Timestamp, quality, value)
- AllenBradley, Siemens, National Instruments, ...
  provide OPC Server as a generic “Driver”

OPC and CA

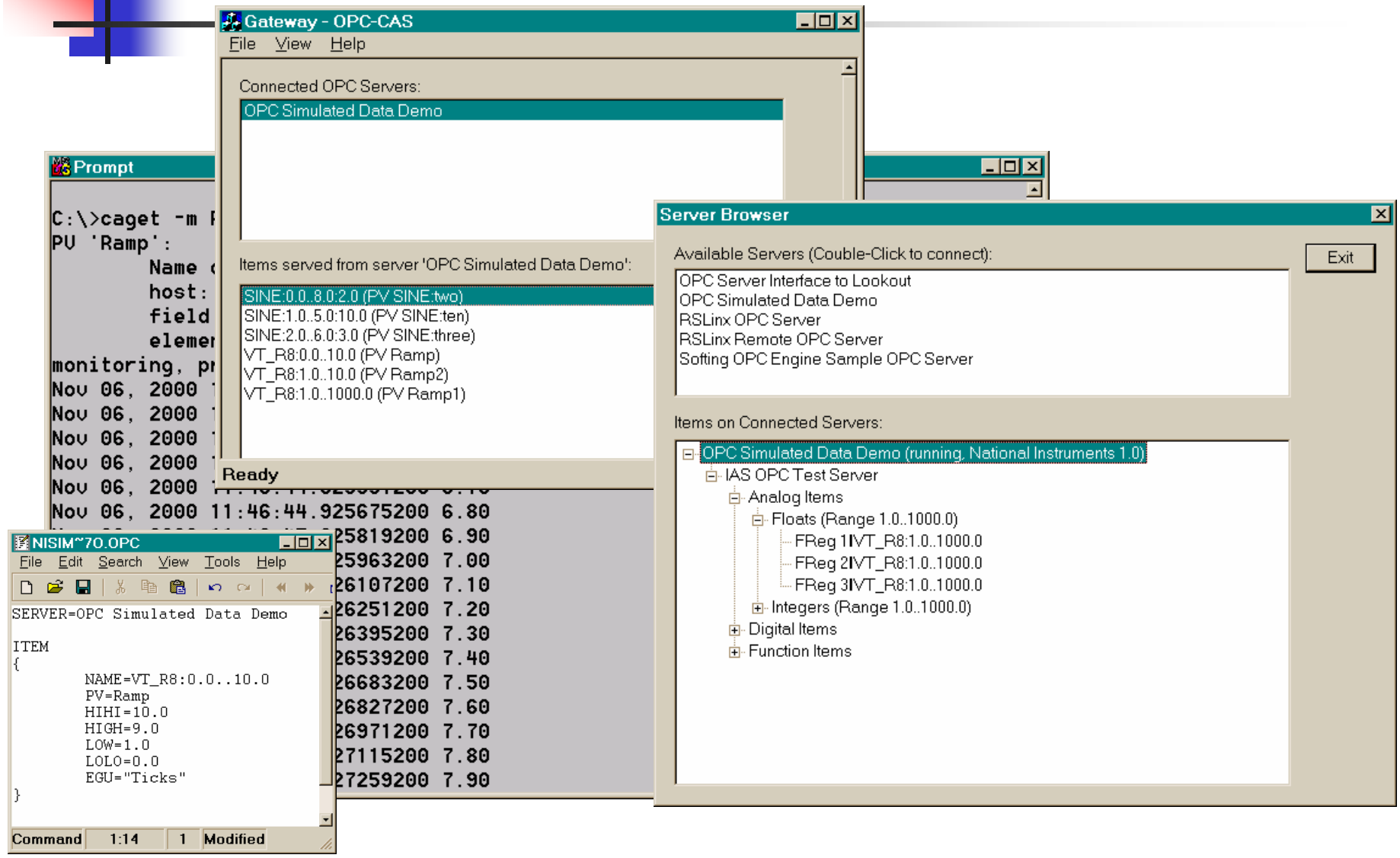
- OPC-CAS: OPC client, CA Server
  - allows browsing (mostly for debugging)
  - reads ASCII config file
  - puts all Items in one group “Default”, requests 10Hz updates, serves via EPICS ChannelAccess

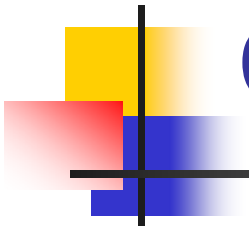
```
SERVER=OPC Simulated Data Demo
# Explicitly list OPC Name & PV:
ITEM
{
  # Required: Name of OPC Item
  NAME=SINE:0.0..10.0:2.0
  # Optional: PV, defaults to Item name
  PV=Sine
  # Optional, will be served/observed
  EGU=Ticks
  HIHI=9
  HIGH=8
  LOW=2
  LOLO=1
}
```

```
# Browse OPC Server, pick OPC items via regular
# expression, maybe use translation file.
# Here, items matching “SINE:xxx:yyy”
# will be served as “SINEzzz” where
# “zzz” is “yyy” translated via numbers.trs which
# looks like this:
#    1.0   one
#    2.0   two
ITEMCLASS
{
            NAME=(SINE):.*:(.*)
            TRANSLATE2=numbers.trs
            DELIM=:
            EGU=Parts
}
```

- Andrei Makhnachev (DESY) wrote CA client/OPC server
- Both: Softing OPC lib. $\Rightarrow$ can distribute binaries, needs license to compile

OPC-CAS Snapshot





Conclusions?

- OPC: Great Idea for PC platform
  - Standard across vendors, no longer separate drivers for LabVIEW, Intellution, VisualBasic, ...
  - EPICS 3.14 for Win32 Hosts should support DTYP "OPC"
    $\Rightarrow$ benefit from OPC drivers instead of having to implement many device specific drivers

```
record(ai, "opc:xx")
{
   field(DTYP, "OPC")
   field(INP,"# 'OPC Server XYZ' fred")
...
}
```

- Problems:
  - No experience with performance, dependability
  - Already different versions (IOPCAsyncIO2...)
  - Networked via DCOM but no connection management