Poster presented as TUP46G at EPAC 98, The Sixth European Particle Accelerator Conference
Stockholm, 22-26 June 1998

ConSys - A New Control System for ASTRID and ELISA

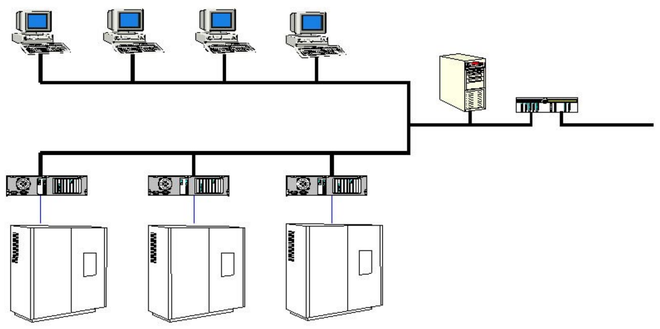
J.S. Nielsen, K.T. Nielsen, and T. Worm
ISA, University of Århus, Denmark

Introduction

The development of the new control system for ASTRID, ConSys, was started at ISA in the beginning of 95. The aim of the project has been to develop a site and machine independent control system based on modern tools and design ideas. Although UNIX workstations is widely used for control systems elsewhere, ConSys is developed for Windows NT on PC's. A major advantage by this choice is the enormous amount of commercial software and hardware for the PC platform. Furthermore, the Windows NT user interface is well known by most users.

Hardware Model:

- TCP/IP network
- PC's running Windows NT all over.
- Main IO: G64/PCDoct

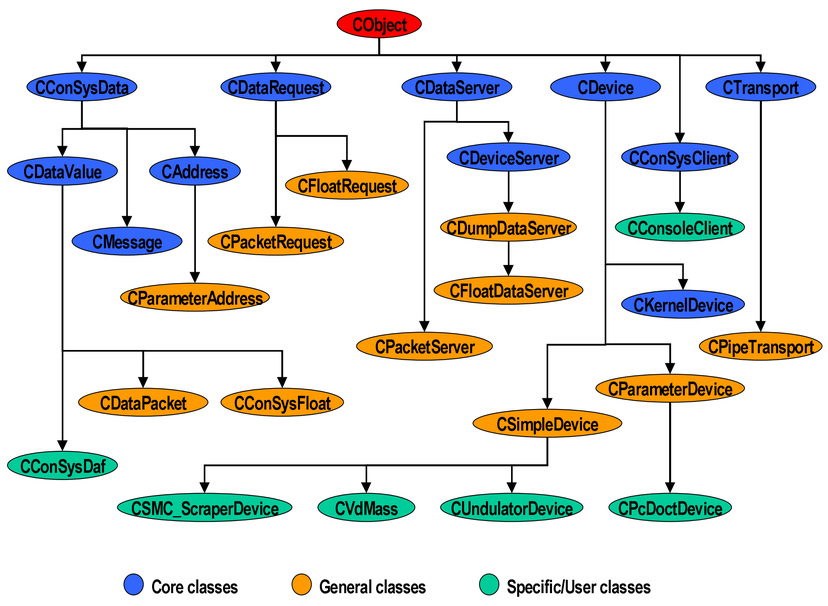


The system is based on the standard model for present days control systems - an Ethernet based system with distributed front-ends and console/client computers. There is a central server for domain management, file server and configuration database. The control system is connected to the in house network. Since front-ends as well as console computers are PC's running Windows NT, it has been possible to use the same control system core software on all computers.

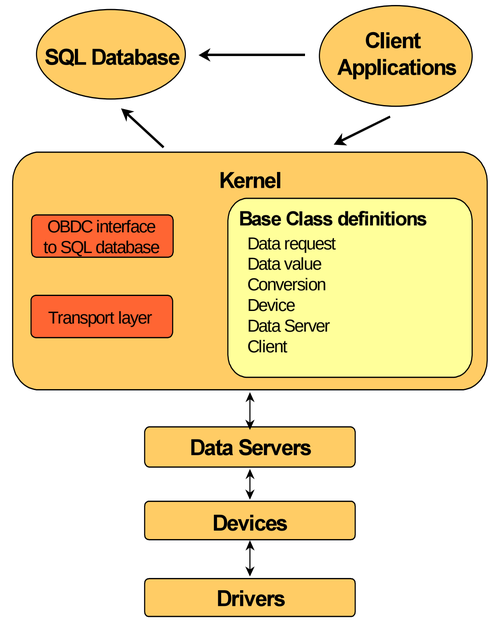
Software Model:

- Publisher/Subscriber
- Object oriented data structure

When a client program wants data from or control access to a parameter on a device, it first has to subscribe to the parameter. The system works much like subscribing to a newspaper: When there is a new value/ newspaper available, it is delivered to the subscriber. The subscriber can send comments/commands to the publisher, which then takes the proper action.
A total object oriented approach of the software design - with a high priority of code reuse and flexibility - has been adopted. Central objects are: address labels, requests, data values, data servers and devices.



Software Components



Database:
- Microsoft SQL server, containing all machine dependent information

Clients/Applications:
- Applications for machine operation, like the general-purpose console.

Kernel:
- The same on all computers.
- Includes: Transport layer, Interface to the SQL database, Base class definitions for all ConSys components.

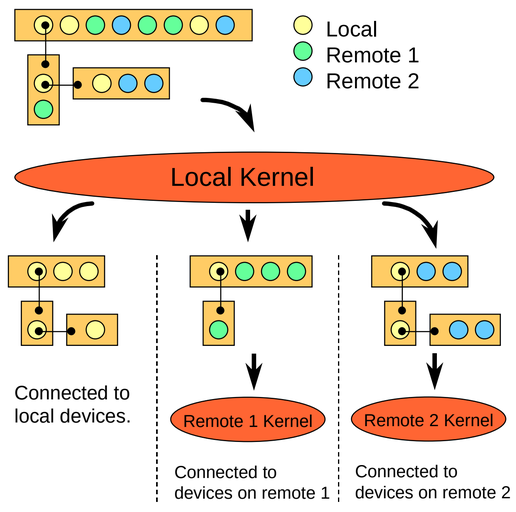
Data Servers (Acquisition Agents):
- Transmitting data objects between devices and clients.
  Each parameter registered at a device has its own data server. Whenever the device has new data it triggers the data servers registered at the device. Based on the initial request from the client, the dataservers decides whether the data should be transmitted to the client or not.

Devices:
- The interface layer between ConSys and hardware devices.
  Has a generalized interface for client registration and data IO to the control system.
- Storage of the parameter values.
  It is in the devices the values of the parameters are remembered. A special subset of devices is the virtual devices, which do not have direct hardware access. They instead serves as storage for information parameters, or for intelligent data manupulation/reaction (for instance calculation of lifetimes, or automatic choice of multiplexer values, as viewers or cups are taken out of the beam. It is also an important feature of ConSys, that devices also can be a client, i.e. subscribe to any parameter on the system.

Drivers:
- NT drivers for hardware devices.
  The G64 units used as primary IO for ASTRID are interfaced through a new PC card, the PC-Doct, developed at the IFA computer department. A Windows NT driver has been developed for this card. Most new hardware bought in the future will include NT drivers.

Creating a subscription

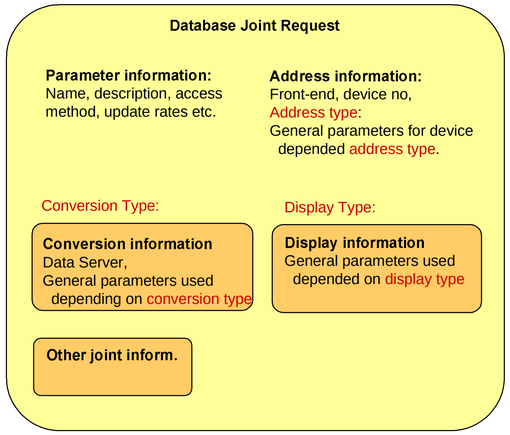
- Client: Constructs a request
  At present, two request types exist; A Parameter Request which is a request for a single parameter on the system, and a packet request, a request containing a list of Parameter Requests. The basic request includes the address object needed to locate the parameter, and specifies the dataserver object to be used with the request.
- Client: Sends the request to local Kernel
- Local Kernel: Splits the request into front-end depended requests
  When the Kernel receives a packet request, the requests it includes are sorted by front-end name and placed into separate requests for each front-end. If a packet request is for the local machine, the request is processed locally as described below for the remote front-end. Other requests are forwarded to their respective front-ends.
- Front-end Kernel: Creates data servers and registers these at the specified devices.
  When a request has reached the front-end computer specified in its address object, it is ready for the final connection. First, the dataserver object specified in the request is created. The dataserver then registers itself on the device addressed by the request. Finally, the dataserver establishes a connection to the transport layer for communication of data between itself and the client.
  A special dataserver, the packet server, handles the packet request. The Packet Server collect data value objects from the dataservers created by the packet request. The received data values are stored in a Data Packet object, a special data value object that can store a list of data values. The packet server sends the data packets to the client at rates given by the current workload on the system and conditions set by the client. Devices can signal data servers to send data immediately.
  The collected data values objects are stored in a Data Packet object - and the corresponding type for data transmission, the data packet. All data transmitted between the client and the device inhered from a common class, the data value class. Data packet, are like Packet requests, a list of data values.



Database

- Used to store all machine/hardware dependent information - including information needed for client programs to generate displays.
- Any ODBC database can be used. We use the Microsoft SQL server.
- Initialization data for ConSys objects stored in generalized, flat tables

The database is build by a number of related tables. The object-oriented structure of the control system cannot be represented directly in a relational database. Therefore, many tables representing a class hierarchy has some common fields used to initialize the base class and a number of generalized fields to initialize the inherited objects. A number of helper tables describe the actual meaning of the generalized fields for a specific ConSys class. With the aid from the helper tables an object oriented database editor has been created.



Most ConSys requests to the database use the same joint parameter request, build from a series of central database tables. The joint data request includes almost all information available for a given parameter, like addressing, interpretation/conversion information, data type information and display information.

Front-end hardware

- PCDoct/G64

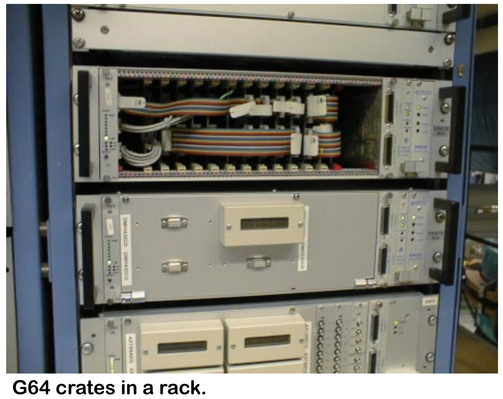
  Most parameters are controlled though G64 (small-scale bussystem). Each G64 communicates with a PCDoct card with a 19200 baud serial connection.
- PCDoct

  The PCDoct is a homemade (by the IFA computer department) PC card with an onboard 80186 processor, which serves up to seven G64's. To interface with high level code a Windows NT device driver has been developed.
- G64

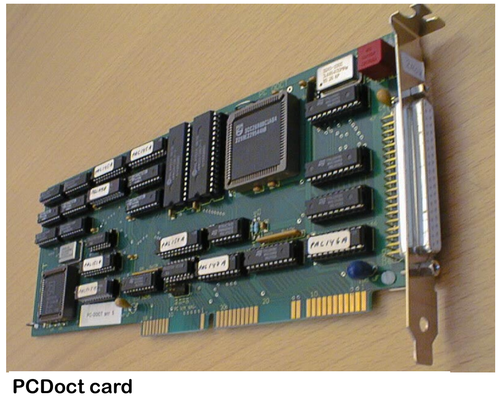
  Original a CERN standard the IFA-G64 system has been further developed with new IO cards. The system consist of a Z80 processor (4 MHz) communicating with up to 10 IO cards via a 8 bit backplane. The IO cards available at IFA are:

  Digital IO card (32 bit individuable in/out, optocoupled/direct),
  ADC cards (1x16 bit or 16x12 bit),
  DAC cards (2x16 bits or 14x12 bits),
  DAF (Dfi Autonomous Functiongenerator) card: up to 512 vectors, update time: 100 µs externally synchronized, minimum vector time: 2 ms, maximum vector time 300 h,
  FAF (Fast Autonomous Functiongenerator) card: Specially developed IO card with DSP processor for RF frequency calculation from bending dipole field.
  Frequency counter card: 16 bit, 0.01 s, 0.1 s, or 1 s gate time
  Puls generating cards: Several cards with different resolutions, user configurable.
- Other hardware.

  Other hardware can be divided into two groups: Instruments communicating with a standard interface (Serial, GPIB, etc.), and instruments which are or require special PC cards (PCI or ISA). For PC cards a Windows NT driver is needed, which is the case for most new PC cards. For standard interfaces, there exist many good products, which also support Windows NT. We have chosen to use National Instruments products for many of these applications, because we have found these products to work well under NT and they have good support.



G64 crates in a rack.



PCDoct card

Commercial Tools

Policy: Commercial tools used whenever possible:
+: Fast development cycle, professional look.
- : Limitations (bound to one solution).

- Microsoft Visual C++ 5.0

  The primary development tool. The control system has a strong bound to MFC, Microsoft Foundation Classes. The communication protocol is based on pipes, an intrinsic part of Windows NT.
- Microsoft SQL server

  Used to store all configuration data.
- Microsoft Excel

  Used a editor for the DAF-files (function tables).
- Component Works (National Instruments)

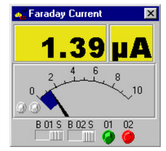
  Active X controls, used for plotting and technical controls.
- CVI (National Instruments)

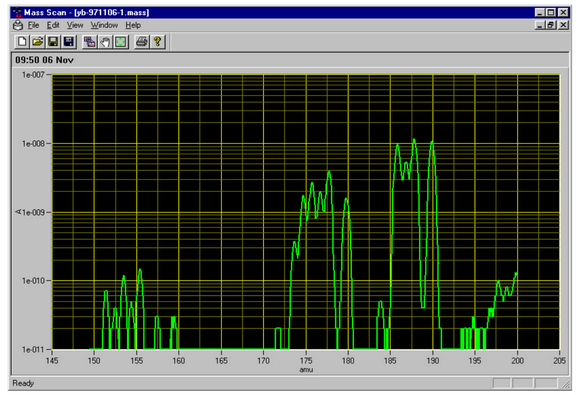
  Can be compiled into the ConSys code. Used for device drivers and some displays. Also used as mathematical library.
- WWW Browser

  Used for documentation, and for status display. In the future possibly also for control (Java).
- Visual Basic, Excel, Borland Delphi Pascal, LabView etc.

  Is supported through the ConSys API (a Windows DLL) for user programs.

Examples of applications using Component Works:





Status

- System is running with most of our parameters
  - Most of the G64 crates without DAF's are running under ConSys.
  - The system has been testet with all parameters (except for FAF-load (RF-frequency ramp)) and will be deployed for next El-run starting beginning of August 1998.

    For Ion-runs we are waiting for an update of user datataking software to work with ConSys. This is expected to be ready for the next Ion-run starting end of October 1998.
- Completed
  - System core (Transport, data structures, database, definitions etc.)
  - PC Doct card & device driver for Windows NT.

    Twentyone G64 units are controlled by ConSys. (Fifteen still on old control system)
  - Virtual devices for ASTRID and ELISA diagnostics implemented.

    Lifetime calculations, current calculations, automatic selection of input (cameras following viewers, Faraday current measurement following cups), etc.
  - All parameters on ASTRID are defined and can be controlled through the old control system.

    2602 parameters defined divided into 750 groups. (One group is typically one supply.)
    4 dedicated frontends, 3 combined frontend/client computers, 6 dedicated client computers, and a number of office computers.
  - General purpose Console.
  - DAF load (function generated load)
  - An API interface allow general Windows programs to connect to the control system (a Windows DLL).
  - Other applications:

    Database editor, Data logger, Soft ramps, Log viewer, Store/Restore of control settings, Beam plot (CVI appl.), Tunechange program, Simpel Web Writer, Small utility programs.
- Missing
  - Important auxiliary applications.

    General plot program, Orbit correction, Alarm and Surveillance program.
  - Few auxiliary system parts.

    Conversions for vacuum gauges, "one to many" virtual device, Photon-energy control (setting) of Undulator, Other small devices.
  - Improved robustness.

    Autoconnect of clients, small memory leak.

Experiences/ Conclusion

- Cost/Development time
  - Project design considerations: 1 man-year
  - Kernel & Database design/impl.: 1½ man-year
  - Applications: 2 man-year
- Performance

  Good: Data rates limited by the network - or in practice, by display capabilities on Console PC's

  11 Console PC's connected to 1 Front-end PC, 125 Clients, each having 8 parameters updated with 4 Hz <=> Front-end CPU 50 %, network 10 %.
- Stability
  - The Windows NT domain: Very good
  - Individual Windows NT workstations: Very Good
  - SQL Server: Initially, some problems with network,settings.
  - ConSys: Reasonable;
    - Front-end with PCDoct: Good - never down.
    - Console PC's: Applications may fail - but the kernel on the PC survives in most cases.
- Limitations
  - Strong bounds to Windows NT & MFC.
  - For the moment no interface to other platforms.
  - Not drivers for all hardware (especially old hardware)

More information

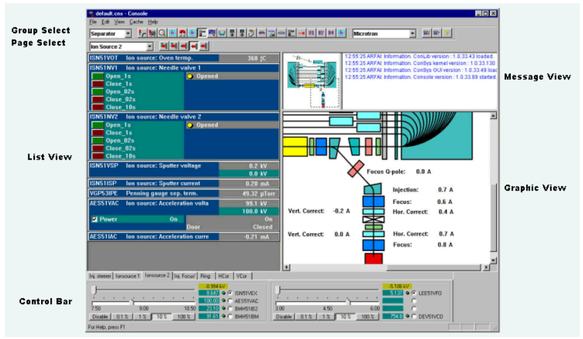
- http://isals.dfi.aau.dk

  This is the homepage of the the ConSys system. Here you will find all the online documentation.
- http://isals.dfi.aau.dk/pdf/EPAC98ConsysPoster.pdf

  This poster.
- http://isals.dfi.aau.dk/pdf/EPAC98ConsysArticle.pdf

  The accompaning EPAC98 article.

Examples of client programs



Console

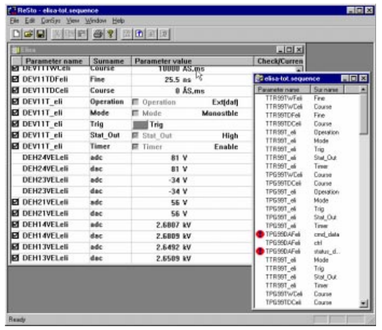
Main program to display and adjust parameters.

- Machine independent - all console pages are fully defined in the SQL database.
- Two control bars with “analog” display of two parameters.
  Each one selectable from a list of 4 parameters.
- ‘Analogue touch’ through digital potentiometers.
- Mixed graphic and text based selections.
- Drag & Drop of parameters to control bars or other ConSys applications.

Digipot

To achieve an “analog” feel when controling the parameters, each console computer has a box with two digital potentiometers, which can be used to adjust the two active parameters in the Console control bar.

- Works by sending messages to the active window.
- Connects directly to a standard serial port.
- Can be installed as a service.
- With buttoms for push and pop parameters, and for selection of active parameters.



ReSto

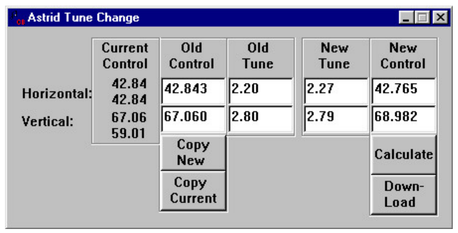
A general purpose program to store and restore parameters on the system.

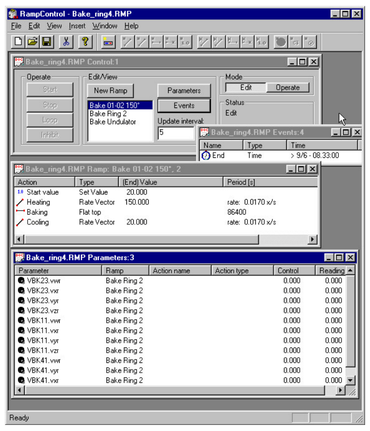
- Upload parameters from one or more machines (or machine parts).
- Download parameters specified in a sequence file.
- Possibilities for sorting parameter names or values.
- Possible to upload new parameters and compare with already uploaded values.

ASTRID TuneChange

A specific program to assist in tune changes.

- Up- and downloads of quadropole currents.
- Knows the magnetic layout of the ring.





RampControl

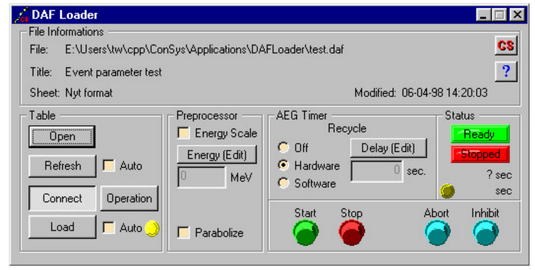
A general purpose program to perform (slow) ramping of parameters.

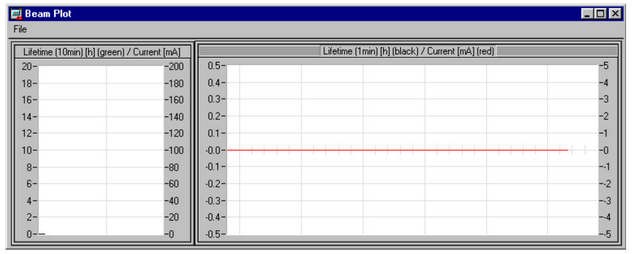
- Ramp floating point values in either specified time or at specified rate.
- Can also set binary values.
- Can wait for specified events. (reached a given time or a parameter has reached a given value (either floating or binary values)
- Rates up to 1 Hz.

DAFLoader

A general purpose program to load DAFs (Dfi Autonomous Functiongenerators).

- Reads files generated in Excel (or any other suitable program).
- Can automaticly load new tables if datafile has changed.
- Possibilities for preprocessing of the data. (Parabolize bends and small energy scaling.)
- Buttoms to start and stop cycle.
- Display of vectortime.





BeamPlot

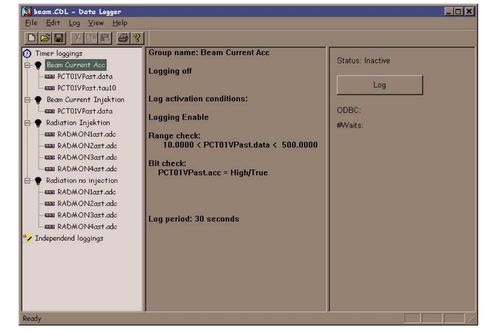
A specific program to plot beamcurrents and lifetimes during electron runs.

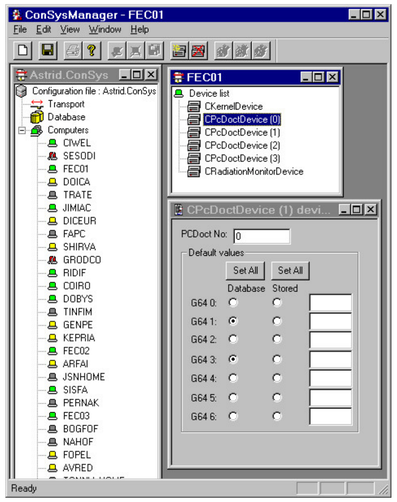
- Three windows with different timeintervals.
- To be replaced by a general plotting program.

Datalogger

A general purpose program to log run-time data values to the SQL database.

- Logging conditions (binary and floating point ranges).
- Rates up to 1 Hz.
- Data format simple triplet (time, source, value)
- Start and stop tacks added to log





ConSysManger

The system setup program.

- Specification of system database and communication protocol.
- Specification of which computers participate in the system.
- Specification of which devices to be loaded on participating computers.
- Setup of devices on the participating computers.

Database Editor

A system program to ease setup and maintebance of the system database.

- Object oriented editing of the ConSys tables - based on description tables.
- Ease maintainance and secure data integrity.
- Used together with Access for editing ConSys SQL data.