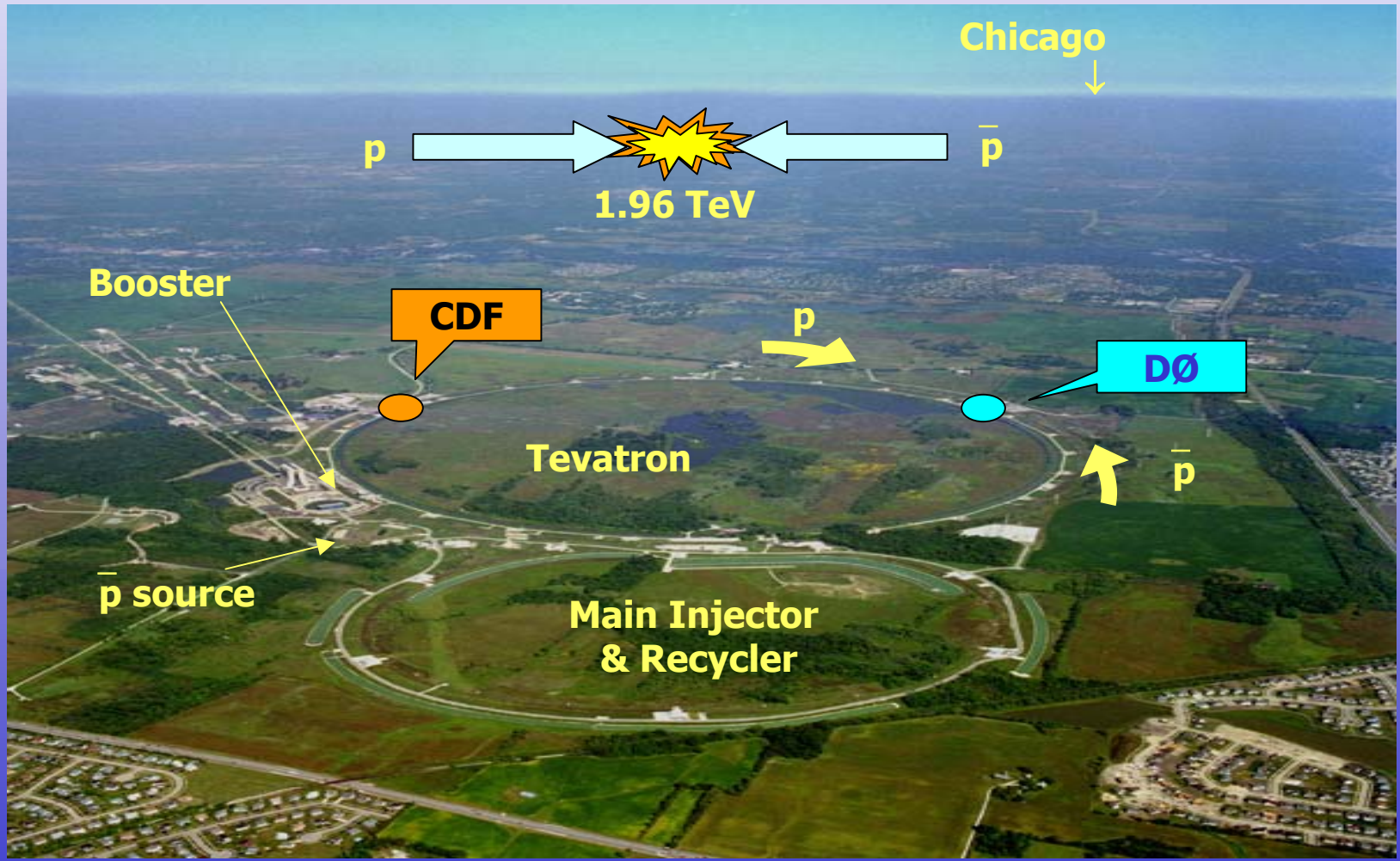
Control and Monitoring of the DZERO Detector at Fermilab

Geoff Savage for the
DZERO Controls and
Monitoring Group

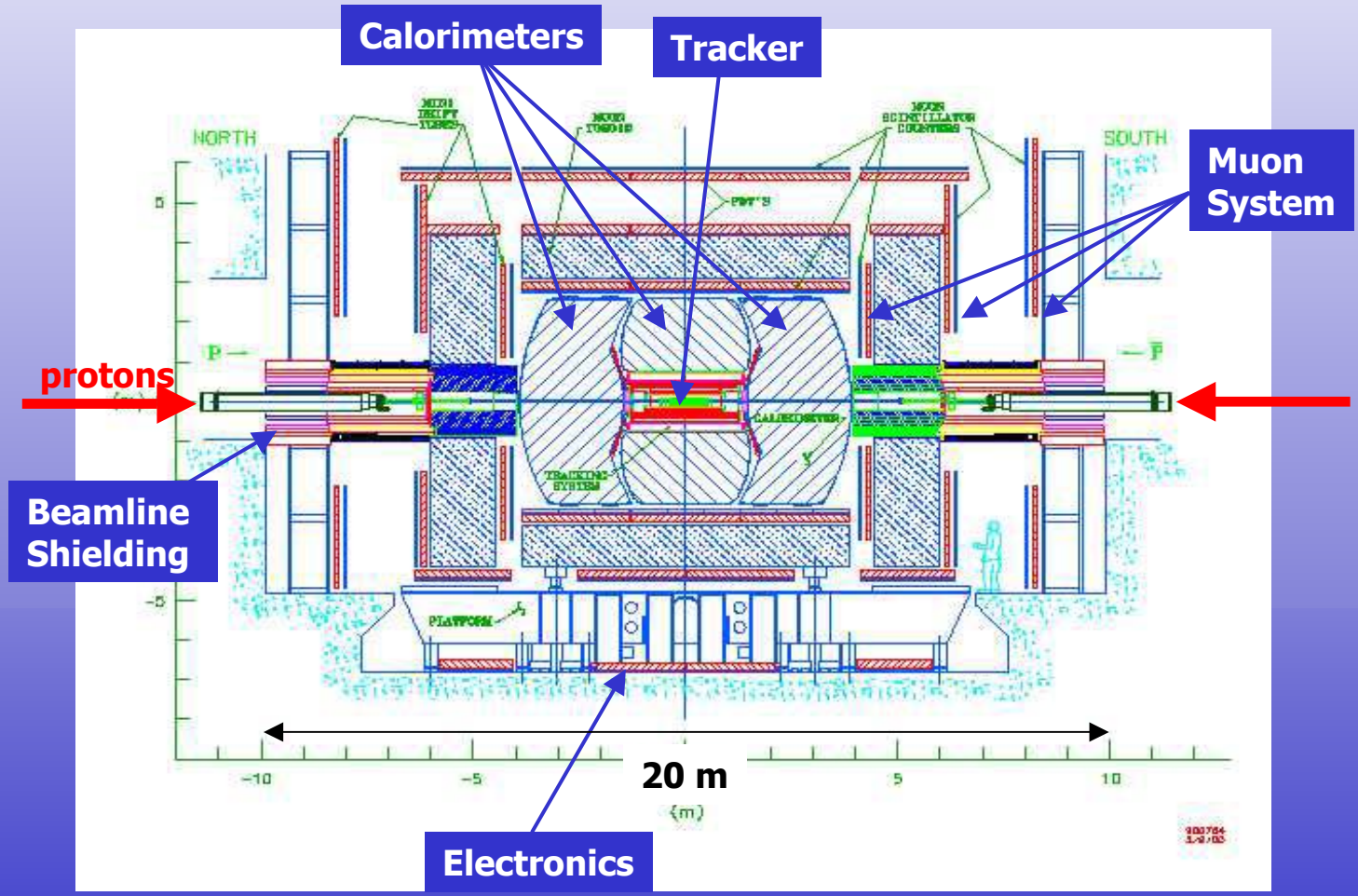
Outline

- Fermilab and DZERO
- EPICS at DZERO
  - New field bus and devices
  - Centralized hardware database
  - Connection to the DZERO alarm system
  - Channel access from Python
  - Host tools for downloading and monitoring
  - Portable channel access server
  - Archiving
  - Infrastructure
- Short term development
- The future of EPICS at DZERO

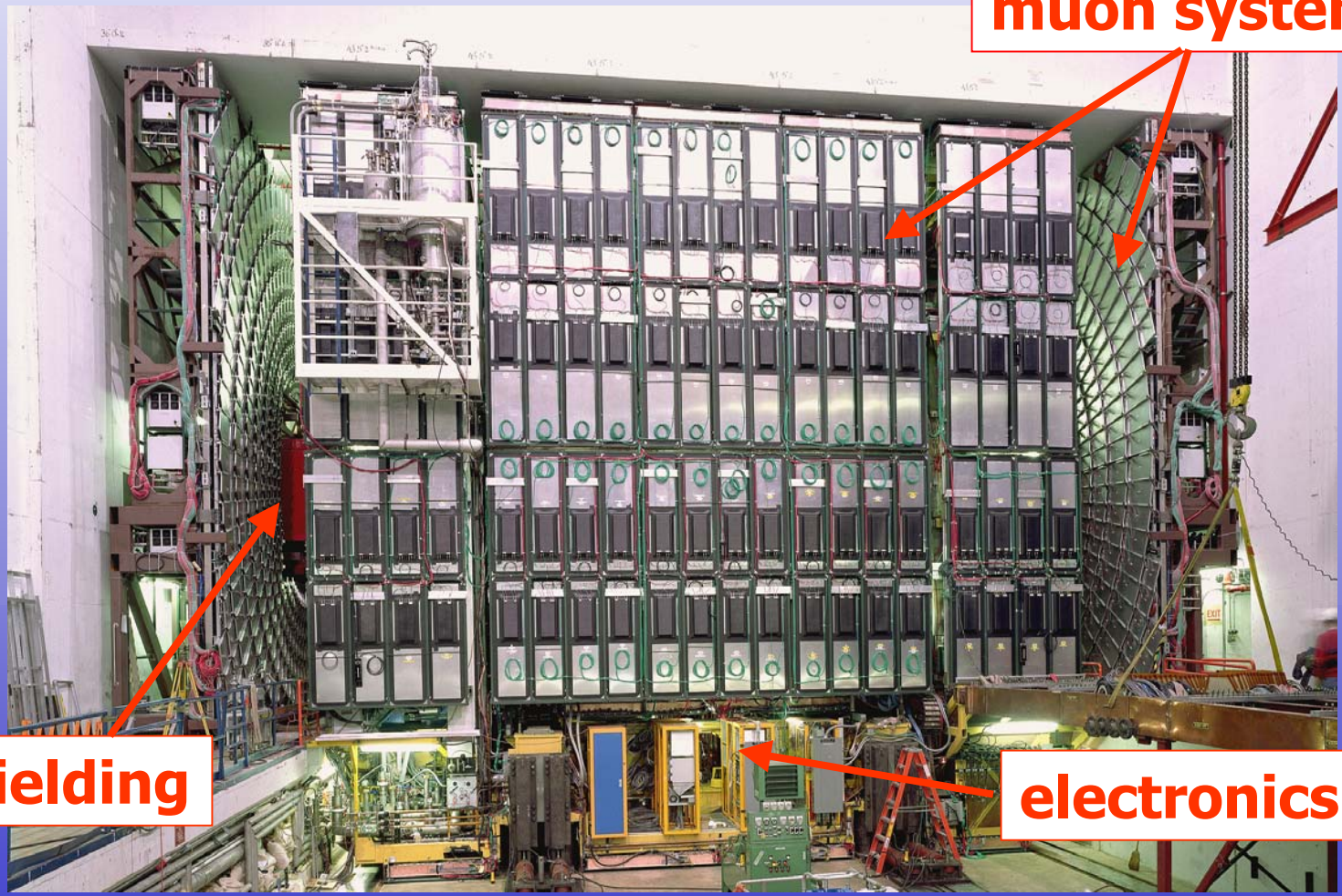
The Fermilab Site



A Cross Section of the Detector



As Big as Your House



A New Field Bus – MIL/STD1553B Serial Bus

- Restricted detector access while running
- Provides a robust and highly reliable connection to electronics in the remote collision hall
- 12 controls crates with ~70 1553 busses from the counting house to the detector and ~10 busses within the counting house
- Developed a queuing driver, device support, and a generic record

DZERO Specific Records

- High voltage
  - Implemented using a state machine model
  - Linear ramping with retries
  - Trip condition recovery
  - Limits control
- SVX sequencer – configures and controls the ADCs in the tracking detectors
- Rack monitor
  - Generic I/O module
  - 64 A/D inputs, 4 D/A output, 4 16-bit I/O words
  - Environment monitoring
  - Low voltage power supplies
- Mil1553 – debugging/testing on a 1553 bus

DZERO Specific Device Support

- Register access records (ai, ao, li, lo, wf, …) through the Mil1553 bus and VME backplane
- DZERO specific records
  - Bira 4877 power supplies
  - SVX sequencers
  - Rack monitor
- Complex 1553 bus and VME backplane access for register access records
- Read-modify-write for mbbo soft raw

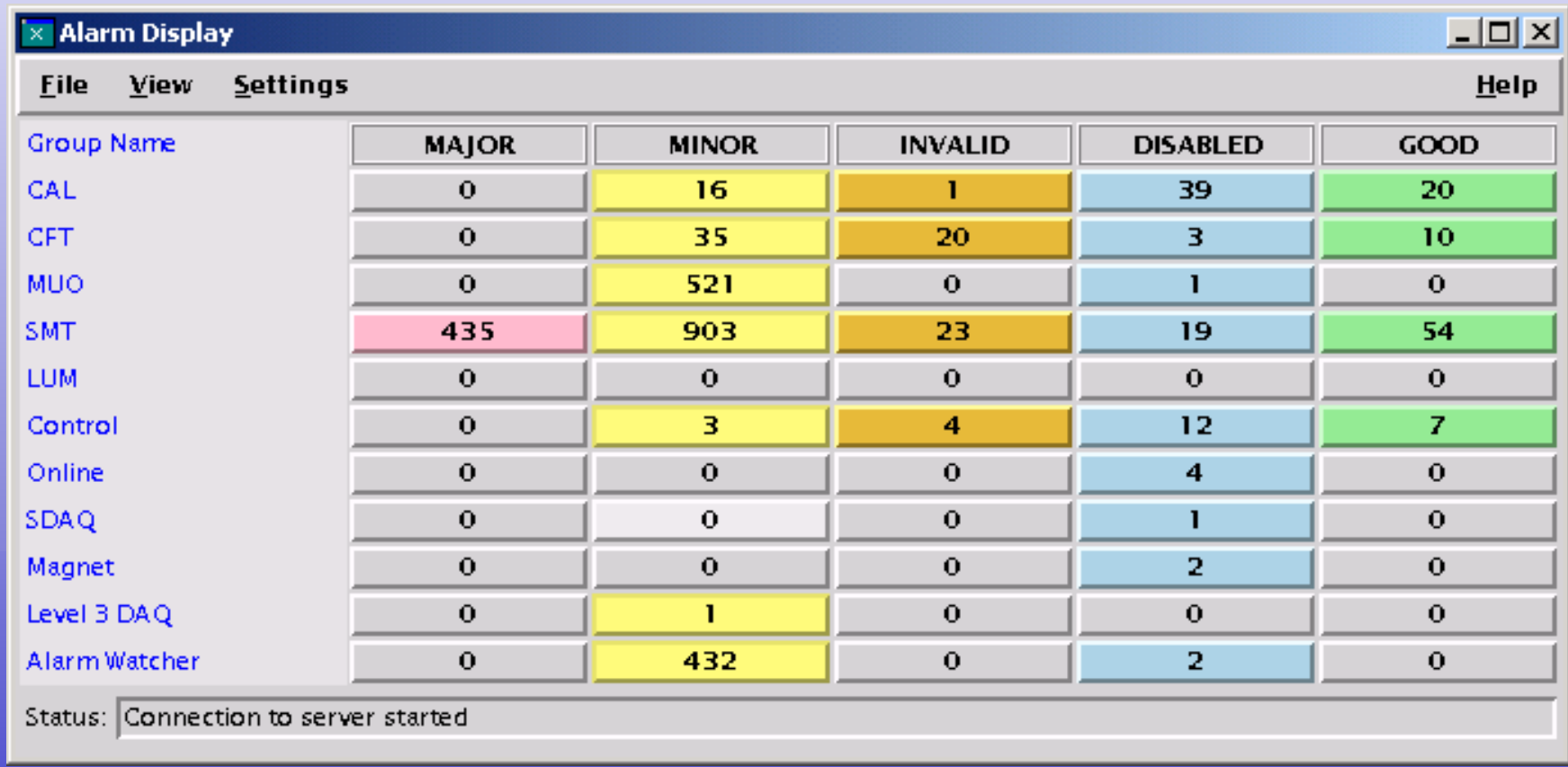
Centralized Hardware Database

- EPICS databases are generated from information stored in an Oracle database
  - Templates
  - Generator files
- A collection of Python scripts exists for bi-directional conversions
- A WWW browser interface is also available
- Some database statistics
  - 195 templates representing devices
  - 4940 devices
  - 123,486 records
  - 1,1161,659 fields
  - 52581 macro values

Connection to the DZERO Alarm System

- Alarms at DZERO are handled by the significant event system (SES)
- An alarm is a significant event
- On a state change in a record an alarm message is sent to the SES server
- The server holds the current state of the experiment
- All significant event messages are stored in a log file on disk
- Alarms are shown on the Alarm Display
- There are no configuration files to specify which records to monitor

DZERO Alarm Display



Using Channel Access from Python

- Python wrappers for CA functions
- From the functions create a class (CaChannel) so that CA is “object oriented”
- Forms the basis of all communication with the IOCs

Monitoring the Detector

- High voltage
- Low voltage
- Environment
- Expert GUIs

Downloading the Detector

- COMICS
  - Puts the detector in a specific state
  - See Fritz's talk tomorrow
- Expert download GUIs for each sub-detector used during the commissioning phase of the detector
- The expert GUIs will continue to be used for calibration and testing

Portable Channel Access Server

- Receive information from other systems
- DZERO cryogenic and gas systems
  - Windows NT host
  - Uses SCADA based DMACS/IFIX32
- The hall probes monitoring the field in the solenoid
  - Window NT host
  - CANbus from the host to the hall probes

Archiving

- Each sub-detector runs it's own channel archiver
- Archives are viewed with striptool and the web interface
- Developing a "slow" (every 15 minutes) archiver that enters data directly into an Oracle database
- The "issue" with the database is how to backup the data

Infrastructure

- All embedded processors run VxWorks
- Our EPICS version is R3.13.4 with modifications for the DZERO alarm system connection
- Our current controller list includes:
  - Mv162 (40) – muon
  - Mv2301 (30) – high voltage and controls
  - Mv2304 (25) – readout and controls
  - Mv2603 (1) – radiation monitoring

Short Term Development

- Update what we have already done to work with EPICS R3.14 on VxWorks
  - Conversion of record, device, and driver support
  - Alarm system connection
  - Python CA interface
  - Replace portable channel access servers
- New compiler, Tornado II, means we are moving builds from Linux to Sun
- Other tasks not mentioned here

EPICS Future at DZERO

- Convert to EPICS R3.14 or beyond for the start of RunIIb in spring 2006
- Move to a non-proprietary real-time operating system?
  - Real-time linux
  - RTEMS
- Add an Intel platform?
  - Embedded Intel processor already in use at DZERO in the readout crates
  - Concerned about long term reliability and support of Motorola processors that we currently use

MUON Scintillators – Scientific Sculpture?