EPICS 4 Progress Report

Timo Korhonen (Paul Scherrer Institut)
for the EPICS 4 Collaboration
Outline

• Introduction
• System structures
• Modules
• Data handling and transport
• Interoperability
• Services
• Summary & Outlook
**Introduction**

- EPICS version 3 structure
  - Flat database of records
    - Enables development of lightweight controls applications
    - Combining data into larger entities is cumbersome

- Scientific applications have different requirements than controls applications
  - Data integration facility-wide, from diverse data sources
    - Beam orbit at a certain pulse from distributed front-ends
    - Facility information in relational databases
  - Large data sets with meta-data and values
    - Detector image with dimensions, trigger conditions, etc.
    - AD converter data with sampling rate, bit depth, environment data

- EPICS 4 aims to bring controls and scientific applications closer
  - Structured data support and new network protocol: pvAccess
  - Services for data processing and aggregation
• Systems can consist of
  – Traditional IOCs, talking both Channel Access and pvAccess
  – Services serving complex data, possibly aggregated from different sources
    • Infrastructure services (RDB), model services, live and archive data, etc.
  – Client applications can use either protocol (CA, PVA) for easier migration
    • But only pvAccess can provide complex data
Structure of EPICS 4

• EPICS 4 is a combination of EPICS 3 and modules providing new features
  – New modules on top of EPICS 3 make a version 4 IOC
  – Services that are not IOCs can be programmed using the additional modules

• Single codebase for IOCs and services
  – One set of APIs instead of separate ones

• EPICS 3 infrastructure can be used as is
  – Huge investment in infrastructure that does its job well
  – Re-implementing all that is not realistic for many sites
  – Add what is missing, keep what works well

• In the future, the additional modules will be merged into the EPICS base release
• Modules that make up the base infrastructure of EPICS 4 (at the moment)
  – Build on top of EPICS base release; at the moment 3.14.12 and higher
  – **pvData**: API manipulating of data structures
  – **pvAccess**: network protocol to transport pvData over the network
  – **pvaSrv**: provides to Version 3 records via pvAccess
  – common utilities for the above, example services, etc.

• Specifications and conventions to complement the above
  • Normative types, specification of general-purpose structures
  • pvAccess protocol specification

• See the project website (epics-pvdata.sourceforge.net) for documentation and code
Structured data support (pvData)

- Data entities can be
  - Scalar, array of scalars, structure, array of structures
    - Structures can contain any of the above
    - Top-level entity with a published name is always a structure
  - APIs for structure introspection and data manipulation

```
structure beamOrbit
    alarm_t alarmStatus
        int severity 0
        int status 0
        string message
    time_t timeStamp
        long secondsPastEpoch
        int nanoSeconds
        int userTag
    structure [] positionData
        string bpmName
        double zPos
        double X
        double Y
        double I
```

Top-level structure contains two structures and one array of structures

Structure containing scalars of primitive data types
Data Transport (pvAccess)

• Network protocol to transport pvData: pvAccess
  – Wire protocol for efficient data transfer over the network
    • Even for high-volume data (e.g. pixel detectors)
  – New operations in addition to put, get and monitor (subscription)
    • ChannelRPC: query with parameters
    • PutGet: put-process-get, get back results after doing I/O operation

• Structure vs. data content
  – Client and server exchange introspection information before exchanging data
    • Data on the wire is not self-describing, for efficiency

• Focus on efficiency
  – Transfer large amounts of data
  – Queuing to support reliable data acquisition
Interoperability of Version 3 and 4

• How to deploy version 4 in existing facilities
  – Co-existence of protocols (Channel Access, pvAccess)
    • V3 Channel Access, V4 pvAccess
  – IOCs can deploy pvaSrv to serve record data and metadata
    • and thus become V4 IOCs
  – pvAccess client can use Channel Access protocol
    • No changes to IOC necessary
Interoperability of services

- Interoperability of services and IOCs depends on
  - Talking the same protocols
  - Introspection facilities
  - Knowing what the structures represent

- Normative Types (NT) enable implementation of generic clients
  - Knowing the structure only does not specify what the data represents
  - Define a set of standard structures
  - Specify also what they represent
    - Receiver can handle the data without knowing where it came from

- Services exchange NT structures
  - e.g. NTURI with query parameters
  - Results returned in a NTTable
Services

• Services provide integration of
  – Different sources of data (aggregation)
  – Data processing and manipulation (modelling, conversions, etc.)
  – Facility data, metadata (device lists, device parameters, etc.)
  – Logbooks, utilities,...

• Service-based architecture has several advantages
  – Modular, single source of data
  – Uniformity of communication IOC to facility services
  – Management: internal changes do not affect clients

• Services, existing or planned
  – Channel Finder service provides device views
    • See next talk (TUCOCB05)
  – MASAR: machine snapshot and retrieve (in MOPPC155, Monday)
  – Gather: Collect data from different sources (IOCs, services)
  – Database services: Serve data from relational databases

• And many others, all talking the same protocol
Summary and Outlook

- EPICS version 4 has taken a firm shape
  - base infrastructure for data handling, pvAccess protocol essentially complete
    - Features are being added: Multicasts, access security, etc.
    - Integration into the base is foreseen
- Working groups continue to build on top of v4 facilities
  - Services
    - Modelling, data manipulation, data integration
  - Utilities for facility management
    - Logbooks, etc., that interface directly with EPICS
  - User interface tools
    - Control System Studio interfaces for services, etc.
- Services are being deployed in production
  - Real-life testing brings maturity to the products
- Consult the project website for information about progress and activities
Thank you for your attention!