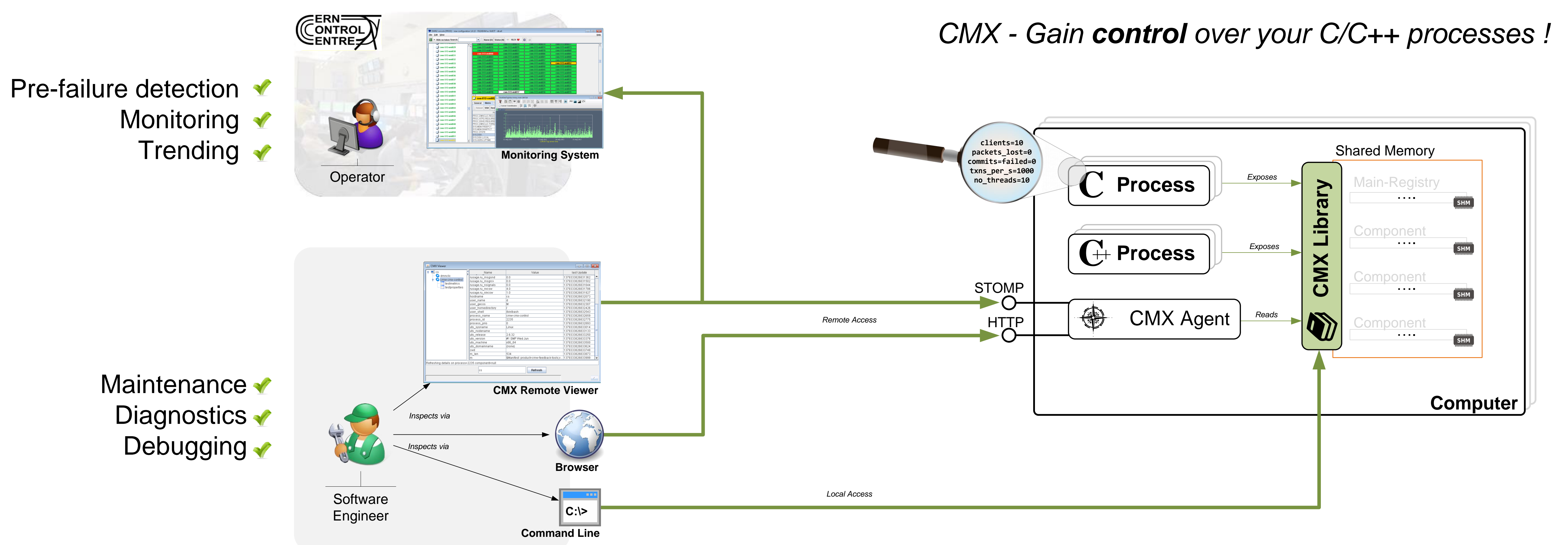


CMX – A GENERIC SOLUTION TO EXPOSE MONITORING METRICS IN C AND C++ APPLICATIONS

Felix Ehm, Yves Fischer, Georgia-Maria Gorgogianni, Steen Jensen, Peter Jurcso
CERN, Geneva, Switzerland

The knowledge of the internal state of processes is essential for problem diagnostic as well as for constant monitoring for pre-failure recognition. The CMX library provides monitoring capabilities for C/C++ similar to the Java Management Extensions (JMX). It allows registering and exposing runtime information as floating point numbers and character data. This can be subsequently used by diagnostics tools for checking thresholds, sending alerts and trending. CMX uses shared-memory for low latency read/update actions, which is an important requirement in real-time processes.

Architecture



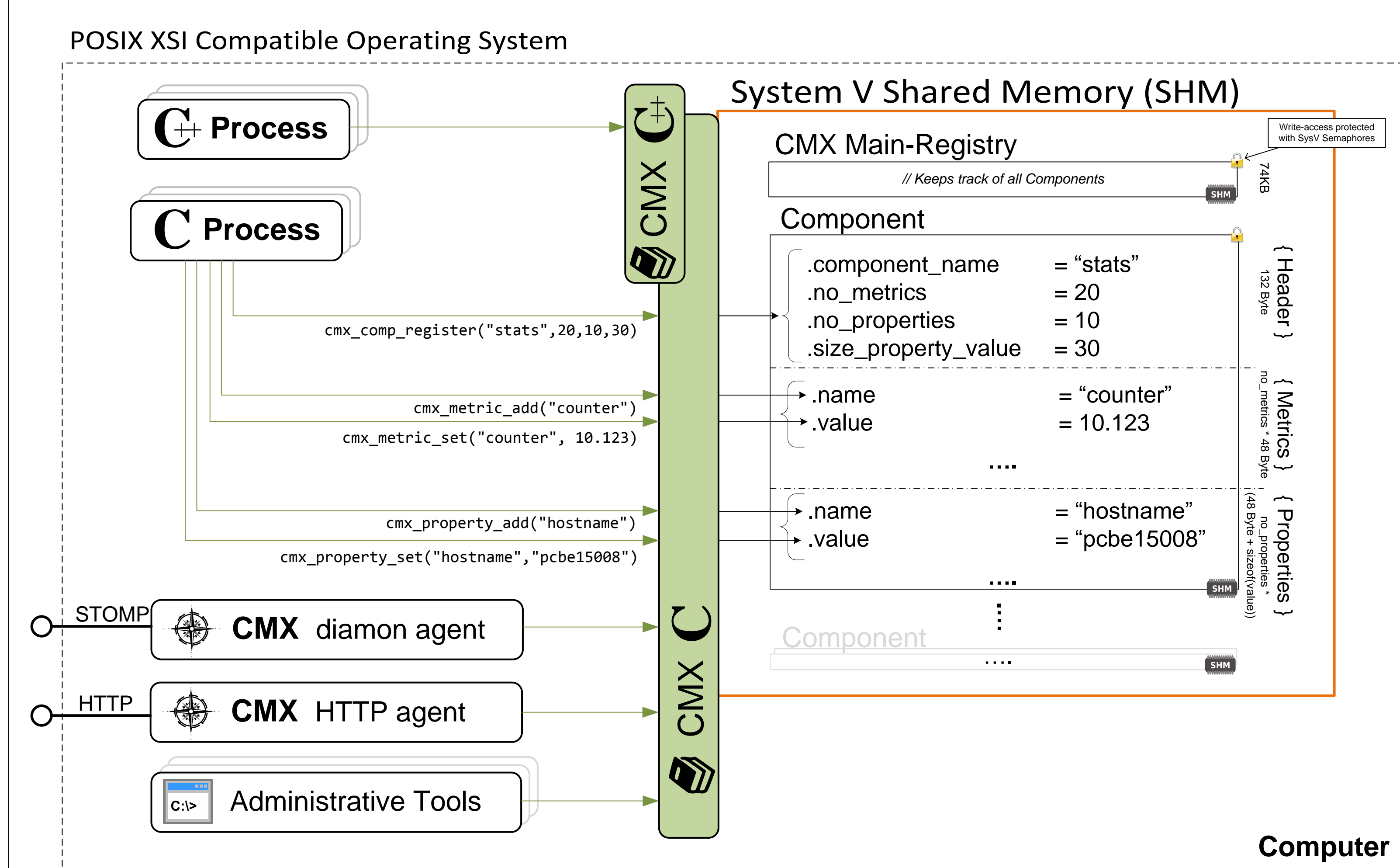
Features

- **Low latency** operations
- Supported data types: **numerical values** and **strings**
- Flexible memory model: number of metrics can be specified
- **Components**: metric groups created **on demand** during runtime
- Automatic **timestamp** for updates

Integration

- CMX has no external dependencies
- **Lightweight**: <100kB code, only 3100 SLOC
- **Simple** and **intuitive** C API
- Object-oriented C++ API available
- **Portable** by POSIX conformance

Implementation Details



Example Use Cases

- Current number of threads
- Thread dead-lock detection
- Network connection tracking
- Transaction per second
- Current memory consumption
- Current working directory
- Current number of threads
- Communication errors

Example: expose build-time information

Manifest Data of an Executable

```

$Manifest: product=p3;version=DEV;... $
$Manifest: depOf=p3;product=p2;version=DEV;... $
$Manifest: depOf=p3;product=p1;version=DEV;... $
$Manifest: depOf=p3.p2;product=p1;version=1.0;... $

```

Is built with p1-DEV

Is linked with p2-DEV

Is linked with p3

Is linked with p1-DEV

Future

- Integration of CMX into the majority of CERN's accelerator control system components
- Use of CMX to monitor about 2000 server processes on 1000 computers of CERN's accelerator control system
- Experience collected in production environment may lead to further extensions
- Elaborate other usage domains and scenarios
- CMX is a **public project**: <http://cern.ch/cmxc>

Conclusions

With the new CMX library, a software developer has a **simple and intuitive** API which offers a time-saving way to **expose internal information** on (real-time) C/C++ processes. For the first time, it is possible to inspect these programs – without using debugging tools – during their execution. CMX is fully integrated into DIAMON, and thus, allows **inspecting information remotely** in the same way as it is now for Java processes using one central interface.

Pre-failure recognition and detailed diagnostics, which are essential for running complex infrastructures, are now possible and the first experiences within CERN's accelerator controls group show that it enhances the monitoring and diagnostic capabilities of C/C++ programs.