

## СРАВНИТЕЛЬНЫЙ АНАЛИЗ АРХИТЕКТУРЫ НЕСКОЛЬКИХ СИСТЕМ УПРАВЛЕНИЯ КРУПНЫМИ ФИЗИЧЕСКИМИ УСТАНОВКАМИ

*Д. Ю. Болховитянов<sup>а,б,1</sup>, П. Б. Чеблаков<sup>а,б</sup>*

<sup>а</sup> Институт ядерной физики им. Г. И. Будкера СО РАН, Новосибирск, Россия

<sup>б</sup> Новосибирский государственный университет, Новосибирск, Россия

Для автоматизации в области физики высоких энергий используется много фреймворков систем управления (СУ). Наиболее распространенными в мировом масштабе являются EPICS и TANGO. В последнее десятилетие все чаще возникает необходимость налаживать взаимодействие (обмен данными) между разными фреймворками. Причиной может быть совместная работа нескольких ранее независимых установок (ВЭПП-2000, ВЭПП-4, ИК в ИЯФ), а также то, что для различных подсистем новых установок могут использоваться готовые решения, уже поддерживаемые конкретными фреймворками СУ. Современная версия системы управления инжекционным комплексом в ИЯФ СО РАН — CXv4 изначально разрабатывалась по модульной архитектуре для облегчения взаимодействия с иными фреймворками. В процессе реализации модулей доступа к EPICS и TANGO был проведен сравнительный анализ архитектур этих фреймворков.

Many control system frameworks are used for automation in high-energy physics. Most common are EPICS and TANGO. In the last decade, a need for interoperation between different frameworks frequently arises. This can be caused by collective operation of previously independent facilities (as VEPP-2000, VEPP-4 and IC at BINP), or new facilities can use solutions, already supported by other frameworks. CXv4, the current version of a framework used at Injection Complex in BINP, was designed in a modular fashion, to ease interoperation with other frameworks. During implementation of data-access modules to EPICS and TANGO frameworks a comparative analysis of their architectures was done.

PACS: 07.05.Hd; 07.05.Wg; 29.50.+v

### ВВЕДЕНИЕ

1980–1990-е гг. были временем быстрого расширения сферы использования электроники, и применение компьютеров для автоматизации физических исследований стало повсеместным. Разнообразие доступной электроники предопределило переход от создания управляющего ПО для каждой установки индивидуально (что становилось слишком трудозатратным) к разработке общих фреймворков СУ, позволяющих составить управляющее ПО из готовых компонентов.

---

<sup>1</sup>E-mail: D.Yu.Bolkhovityanov@inp.nsk.su

Именно в это время были созданы наиболее распространенные сейчас в мире EPICS [1] и TANGO [2], а также трехуровневая модель, по которой построены все современные СУ.

Несмотря на общую базовую модель, EPICS, TANGO, CX [3] и прочие разрабатывались в разных условиях, под *разную* аппаратуру, под влиянием *разных* обстоятельств. В результате, хотя любая из «зрелых» СУ потенциально может быть использована для автоматизации любой установки, между системами есть значительные архитектурные различия, влияющие как на рациональность применения на конкретных установках, так и на удобство взаимодействия с другими СУ.

### ТРЕХУРОВНЕВАЯ МОДЕЛЬ

Все современные СУ построены по распределенной трехуровневой модели, смысл которой в отделении бизнес-логики управляющих программ (верхний уровень) от непосредственной работы с аппаратурой (нижний уровень). Средний уровень является связующим, обеспечивающим передачу данных между управляющими программами и аппаратурой, а также решающим большое количество служебных задач. Тем самым средний уровень является как бы «программной шиной данных». Все три уровня могут располагаться на разных узлах, осуществляя взаимодействие через сеть. Средний уровень обычно именуется «серверным», но корректнее говорить о модели «издатель-подписчик».

Минимальной единицей обмена обычно является «канал» — именованная единица данных, которая может быть атомарно прочитана или записана.

Средний уровень с протоколами общения с остальными двумя уровнями — это и есть ядро СУ.

### О РАСПРОСТРАНЕННОСТИ РАЗНЫХ СУ

EPICS наибольшее распространение получил в США, а TANGO в Европе. Однако, например, европейский ITER выбрал EPICS.

Плюс некоторые крупные центры используют трехуровневые СУ собственной разработки: DOOCS и TINE в DESY (наряду с EPICS и TANGO!), MADOCA на SPring-8. В ЦЕРН — своя СУ под названием FESA, она же выбрана для FAIR.

### EPICS

EPICS был создан в середине 1980-х гг. совместно ANL и LANL. Изначально поддерживалась аппаратура VME с контроллерами на Motorola M68K под управлением VxWorks. Таким образом, вся аппаратура «локальна» (поддержка удаленной электроники была добавлена значительно позже, при помощи модуля AsynDriver).

«Строительными блоками» в сервере EPICS являются виртуальные сущности, называемые «records». Каждый record относится к одному из заранее определенных типов: это аналоговый ввод, аналоговый вывод, цифровой ввод или вывод и т. п. Плюс некоторое количество специальных типов, используемых не для непосредственной работы с аппаратурой, а для организации взаимодействия других record'ов. В простых

и одноканальных устройствах один record представляет все устройство, для более сложных устройств требуется большее количество record'ов. Каждый record содержит некоторое количество «полей» (отсюда и название «record» — как структура данных), одно из которых, VAL, является выделенным и представляет аппаратное значение (например, канала АЦП), а остальные являются свойствами, определяющими поведение record'a.

Совокупность record'ов в сервере EPICS (именуемом IOC — Input-Output Controller) составляют «базу данных», а набор IOC в локальной сети составляет «распределенную базу данных». Управляющие типы record'ов позволяют создавать сложные цепочки обработки данных. Тем самым многие задачи могут решаться без программирования, а просто путем конфигурирования «автоматов» в распределенной базе данных.

Доступ прикладных программ к данным идет через протокол Channel Access (CA), при этом все поля доступны одинаково, без различий между VAL и полями свойств.

В EPICS на уровне прикладных программ и «базы данных» нет понятий «устройство» и «драйвер» — вся работа идет через record'ы, что делает поддержку многих видов устройств (особенно нелокальных и многоканальных) непростой.

С конца 2000-х гг. создавалась новая версия EPICS4, призванная устранить некоторые изначальные недостатки EPICS (в первую очередь — отсутствие структурных типов и атомарности). В настоящее время она, получив название EPICS7 (7 = 3 + 4: EPICS3 с поддержкой протокола EPICS4 rvAccess), эксплуатируется на нескольких установках.

## TANGO

Система TANGO разработана в начале 2000-х гг. на европейском синхротронном комплексе ESRF для замены предыдущей системы TACO (TANGO — TAcO New Generation Objects), созданной в конце 1980-х гг. TANGO изначально была ориентирована на управление подвижками, что видно даже по списку возможных состояний устройств (отражающих возможное поведение именно подвижек).

TANGO — объектно-ориентированная система. «Строительные блоки» в сервере — устройства, каждое из которых является объектом какого-либо класса, унаследованного от DeviceClass. Клиентские программы общаются с устройствами в сервере посредством объектов DeviceProху; тем самым может имитироваться как бы локальный доступ. Вместо каналов единицами обмена являются «атрибуты» устройств (примерный аналог канала) и «свойства»; также устройства могут предоставлять «команды».

В основе TANGO лежит протокол CORBA (атрибуты, свойства и команды — оттуда). Но CORBA, помимо сложности и требовательности к ресурсам, использует *синхронную* клиент-серверную модель, в то время как для СУ требуется асинхронная модель «издатель-подписчик». Это привело к необходимости использования второго протокола — ZeroMQ, и в дальнейшем от CORBA предполагается полностью отказаться.

TANGO конфигурируется через базу данных MySQL, в которой описываются наборы устройств конкретных серверов, параметры их работы и могут сохраняться значения запоминаемых («memorized») атрибутов.

То, что клиент-серверный обмен идет в парадигме «устройство» (от которых требуется особая семантика работы), а не через простые «каналы», сильно затрудняет доступ из TANGO-клиентов к иным СУ.

## СХ

СХ разрабатывался в ИЯФ с середины 1990-х гг. для автоматизации ВЭПП-5 [4]. С самого начала требовалось взаимодействие с устройствами, подключаемыми через разные шины, в том числе САМАС, CAN, RS232/485, Ethernet. Большая часть этих шин не являются локальными, поэтому устройства могут менять свое состояние (включаться, выключаться, сбрасываться) независимо от сервера. Кроме того, большая часть этих устройств многоканальна. В качестве ОС для СУ ВЭПП-5 были выбраны BSDI, а затем Linux.

У коллектива авторов имелся опыт системного программирования под разными ОС, в том числе создания драйверов ядра для UNIX-подобных систем (BSD, Linux). Поэтому была выбрана аналогичная ядрам ОС модель: на нижнем уровне — драйверы устройств с простым асинхронным интерфейсом, на верхнем уровне — парадигма каналов, устройства же на верхнем уровне никак не фигурируют.

Существенно отличается от прочих СУ механизм загрузки драйверов: они являются модулями, подгружаемыми динамически по мере надобности, в то время как в EPICS и TANGO встраиваются непосредственно в бинарный образ сервера (из-за чего требуется сборка отдельного образа для каждого применения и пересборка при любых изменениях в драйверах).

### Сравнение основных характеристик

Характеристика	EPICS	TANGO	CX
Source lang.	C, C++	C++, Java, Python	C
Bindings	C, C++, Java, Python	C++, Java, Python	C/C++, Python
Shell utils	caget, caput, camonitor	—	cdaclient
GUI screens	medm, CSS	TAURUS	pult
OS support	VxWorks, Linux, Windows	Linux, Windows	Linux, BSD
Drv. linking	Static	Static	Dynamic
Device states	—	+	+
Srv.bldg.blks	records	devices	devices
I/O entities	channels	dev: attributes,properties,cmd	channels
Names	Arbitrary string	domain/family/member/atr	device.channel
Aliases	+	+	a.b.c.d.e.{.n}
Case-insens.	—	+	+
Name rslv.	UDP	MySQL	UDP
Server name	—	HOST:PORT/name	HOST:N.name
Type conv.	+	—	+
Net.protocol	Channel Access, pvAccess	CORBA + ZeroMQ	CX
AutoReconn	+	+ / - -	+
Config	Text files	MySQL	Text files
Starter	— (site-specific apps)	Starter + Astor	cx-starter

Также модулями-плагинами являются реализации протоколов связи в клиентской библиотеке и в сервере. Это позволяет налаживать прямое взаимодействие с другими СУ: достаточно реализовать соответствующий модуль. Сейчас есть модули доступа к VCAS, EPICS, TANGO.

### ДАЛЬНЕЙШИЕ ИССЛЕДОВАНИЯ

Во-первых, для полноты картины следует провести анализ как минимум фреймворка FESA.

Во-вторых, сравнительный анализ неполон без сравнения производительности.

1. По скорости передачи больших объемов данных. Тут, вероятно, будет примерный паритет, поскольку определяющей является пропускная способность сети.

2. По скорости передачи большого количества скалярных данных. Предварительное сравнение CXv4 с EPICS3 показывает выигрыш первого.

3. По времени отклика.

Такое сравнение потребует создания под все тестируемые системы драйверов одного и того же устройства, причем устройство должно быть цифровым осциллографом (для генерации большого объема данных). В настоящий момент предполагается использовать крейт NI PXIe с универсальным модулем В/В PXIe-6363 (он имеет как скалярный, так и осциллографический режимы работы).

### ЗАКЛЮЧЕНИЕ

Архитектура СУ является определяющим фактором для возможности организации взаимодействия с другими СУ.

- CX позволяет прямой обмен данными с произвольной СУ.
- EPICS предполагает работу только с EPICS, но позволяет создание «шлюзов» в другие СУ.

- TANGO не позволяет прямого доступа к другим СУ.

Также архитектура фреймворка СУ определяет, насколько эта СУ годится для применения на конкретной установке. Но большинство распространенных СУ являются достаточно «зрелыми» для поддержки любой электроники, а более важно — наличие на установке персонала, способного воспользоваться конкретным фреймворком. Поэтому, на взгляд авторов, именно наличие людей, знакомых с конкретной СУ, должно являться основным фактором при выборе фреймворка для автоматизации.

### СПИСОК ЛИТЕРАТУРЫ

1. EPICS — Experimental Physics and Industrial Control System. <https://epics.anl.gov/>.
2. TANGO Controls. <https://www.tango-controls.org/>.
3. *Bolkhovityanov D. et al.* CXv4, a Modular Control System // Proc. of ICALEPCS-2015, Melbourne, Australia, 2015. P.WEPGF093; doi:10.18429/JACoW-ICALEPCS2015-WEPGF093.
4. *Maltseva Yu. I. et al.* VEPP-5 Injection Complex Performance Improvement for Two Collider Operation // Proc. of RuPAC-18, Protvino, Russia, 2018; doi:10.18429/JACoW-RUPAC-2018-TUZMH02.