

Приборы и Системы.

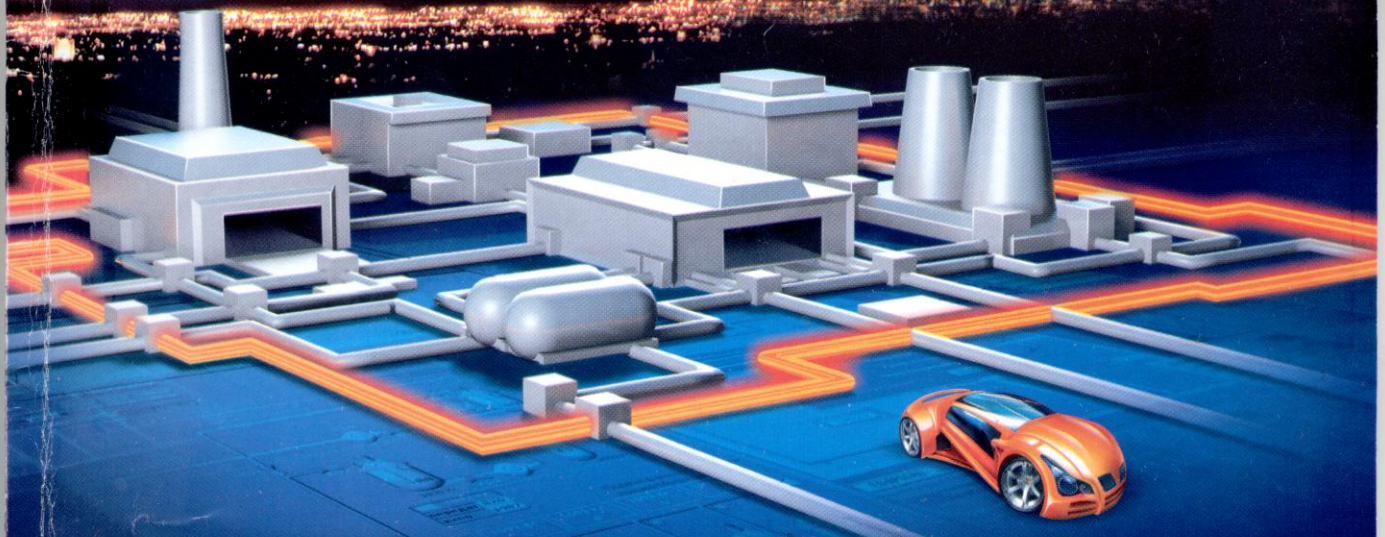
Управление, Контроль, Диагностика

12·2008

Управление производством в системе TRACE MODE

АСУТП/АСДУ/АСКУЭ/СИСТЕМЫ БЕЗОПАСНОСТИ

TRACE
MODE
version 6



XV

конференция новых технологий

Москва, 12 февраля 2009 года. Бизнес-парк Авиаплаза

Добро пожаловать!

Регистрируйтесь сегодня!

www.adastra.ru

Тел.: (495) 771-71-74, факс: (495) 518-98-46
E-mail: expo@adastra.ru



С Новым
Годом!

Организатор конференции:

AdAstra
RESEARCH GROUP, LTD



ПРИБОРЫ И СИСТЕМЫ. УПРАВЛЕНИЕ, КОНТРОЛЬ, ДИАГНОСТИКА № 12 / 2008

ЕЖЕМЕСЯЧНЫЙ НАУЧНО-ТЕХНИЧЕСКИЙ И ПРОИЗВОДСТВЕННЫЙ ЖУРНАЛ

Автоматизированные системы управления

В.П. ВОЕВОДИН, д-р физ.-мат. наук

Управление ускорительным комплексом в среде Linux

В статье дано краткое описание крупнейшего в России ускорительного комплекса *У-70*. Рассмотрены архитектура распределённой системы управления и базовые программные решения в среде *Unix*-подобных ОС, которые в едином информационном пространстве реализуют функциональности, аналогичные современным большим системам *SCADA*. Изложены этапы постепенного перевода работающей системы управления к однородной среде вычислительных средств, совместимых с ПК, и к единственному типу ОС – *Linux*.

The article gives a brief description for the biggest in Russia particle accelerator complex *U-70*. It concerns on the complex distributed control system architecture and on the main software decisions implemented in *Unix*-like environment that allows constructing functionalities in global information space similar to *SCADA* systems. It shows step-by-step transfer for acting control system to uniform *PC*-compatible computing environment operating under *Linux*.

Объект управления и требования к системе управления

Действующий с 1967 г. ускорительный комплекс *У-70* в настоящее время остаётся крупнейшим в России и обеспечивает выполнение программ физических исследований пучками заряженных частиц. Он модернизируется и развивается с целью создания на его базе Центра ионной лучевой терапии. Рассматриваемая система управления (СУ) имеет отношение к следующим установкам ускорительного комплекса *У-70*:

- линейный ускоритель *И-100* с каналом ввода пучка в кольцевой ускоритель бустер *У-1.5* (длина *И-100* и канала, соответственно, 100 м и 41 м);
- линейный ускоритель *ЛУ-30* со своим каналом ввода пучка в *У-1.5* (длина *ЛУ-30* и канала равна 30 и 65 м, соответственно);

- кольцевой ускоритель бустер *У-1.5* с каналом перевода пучка в основной кольцевой ускоритель *У-70* (длина кольца и канала – 100 и 74 м);

- основной кольцевой ускоритель *У-70* с системами быстрого и медленного вывода пучка (длина кольца 1484 м).

Ускорительный комплекс объединяет как ряд подземных сооружений, так и более десятка наземных зданий различной этажности. Последовательность работы установок комплекса состоит из двух этапов.

Первый этап, технологический цикл бустера – однократное срабатывание одного из линейных ускорителей и ввод сгустка заряженных частиц в кольцо бустера, ускорение сгустка в бустере и либо перевод частиц в кольцо *У-70*, либо вывод на локальную экспериментальную установку. Длительность одного цикла бустера составляет 60 мс. Для наполнения кольца *У-70* сгустками заряженных частиц может потребоваться до 29 циклов бустера, в это время *У-70* работает в режиме накопления. После отработки “*пакета*” из заданного числа циклов в работе бустера наступает пауза до завершения цикла *У-70*.

Второй этап – ускорение накопленных в кольце *У-70* частиц и вывод их на экспериментальные физические установки. Длительность второго этапа составляет порядка 8 с, а длительность технологического цикла всего ускорительного комплекса *У-70*, называемого суперциклом, составляет 10 с. Работа ускорительного комплекса – это непрерывная последовательность десятисекундных суперциклов.

На данный момент системе управления доступны для измерения/управления порядка десяти тысяч, в основном, функционально взаимосвязанных разнообразных цифровых и аналоговых сигналов, что составляет примерно 60 % от общего числа сигналов комплекса. Остальные, к сожалению, в силу ряда обстоятельств не имеют компьютерного контроля/управления. Большинство входных аналоговых сигналов многократно измеряется за время цикла установки, а из доступных СУ управляющих аналоговых сигналов около трёхсот требуют изменения выходного значения во время технологического цикла по определённому закону.

По различным причинам, например, из-за изменения радиационной обстановки в какой-то зоне, может потребоваться кардинальное изменение режима работы ускорительного комплекса в ближайшем суперцикле. Система управления имеет для своевременной замены всех значений управляющих сигналов по всему ускорительному комплексу около 120 мс в конце текущего суперцикла, когда по его результатам закончены все радиационные измерения, но до начала следующего. Если для работы *У-70* требуется менее 29 циклов бустера, то *У-1.5* может после обслуживания *У-70* обеспечить пучком заряженных частиц с принципиально другими характеристиками (энергия, форма сгустка, направление вывода) локальную экспериментальную физическую установку. Смену всех значений управляющих параметров технологических подсистем бустера при переходе от обслуживания *У-70* к локальной физической установке подсистема управления бустера должна выполнить за 20 мс.

Контроль и управление комплексом *У-70* осуществляется с главного и пяти местных пультов управления дежурным персоналом служб оперативного управления. Для разработки ПО, проведения тестовых и пусконаладочных работ требуются дополнительные точки доступа к технологическому оборудованию посредством СУ из некоторых технологических помещений и рабочих комнат в различных зданиях. На всех рабочих местах информация об основных параметрах и текущем состоянии элементов ускорительного комплекса должна обновляться каждый суперцикл. Сами ускорители являются объектами изучения, и до 10 % общего времени работы ускорительного комплекса выделяется на его исследование. Исследовательский режим работы комплекса предъявляет дополнительные требования к СУ.

Базовые архитектурные решения

Исторически каждая установка имела свою собственную изолированную СУ, но в 1993 г. были предприняты первые шаги в проектировании единой СУ всем комплексом *У-70* [1]. На рис. 1 схематично показано расположение оборудования созданной системы управления и входящие в комплекс доступные СУ установки на данный момент. Архитектура аппаратных средств СУ соответствует стандартной трёхуровневой модели. Нижний уровень составляют разнообразные контроллеры оборудования (КО) на базе магистрально-модульной системы *Multibus I*, позволяющие удовлетворить наиболее критические требования по времени отклика и реакции системы. Они объединены полевыми магистралями *MIL 1553*. Верхний уровень состоит из компьютеров и интерактивных консолей, подключенных к технологической локальной вычислительной сети (ТЛВС) *Ethernet*, которая по принципам распределения ресурсов соответствует гибридной равноранговой (*peer-to-peer*) архитектуре, т.е. каждый компьютер исполняет роль и сервера, и клиента. Консоли – это аппаратно-программные средства взаимодействия человека с СУ, которые входят в состав пультов управления и являются точками входа в СУ из других помещений. Средний уровень СУ составляли бездисковые каркасы *VME* с одноплатной ЭВМ на базе процессоров *Motorola 68K*, с интерфейсом *Ethernet* и контроллерами магистрали *MIL 1553*. Его основной

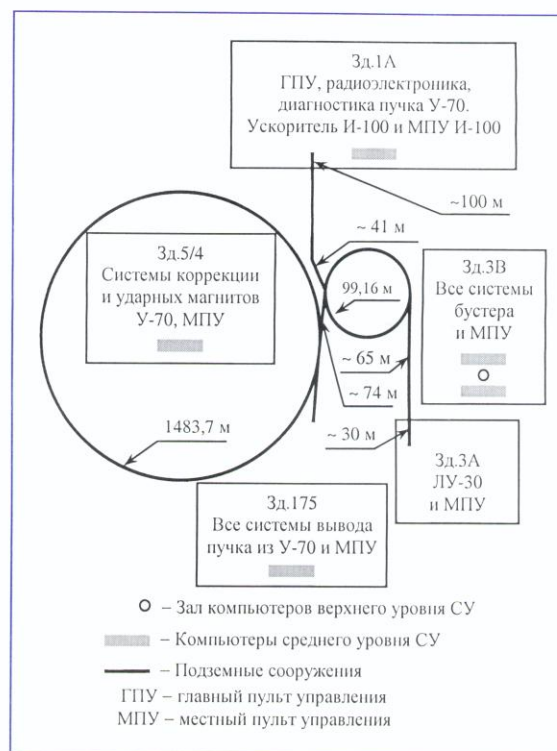


Рис. 1. Схема расположения оборудования СУ в зданиях комплекса *У-70*

задачей является преобразование и передача сообщений между верхним и нижним уровнями СУ по линиям *Ethernet* и *MIL 1553*.

На момент основной закупки вычислительного оборудования требованиям верхнего уровня СУ по вычислительным мощностям удовлетворяли *Alpha* станции, которые и были поставлены. В качестве интерактивных средств управляющих консолей как пультов управления, так и в технологических помещениях, использовались двадцать пять *X*-терминалов с экранами 21 дюйм. Основной задачей верхнего уровня СУ является сбор, обработка и отображение информации на экранах всех активных консолей с частотой, соответствующей времени суперцикла ускорительного комплекса. Программный проект требовал тщательно проработанных концептуальных решений в логической организации ПО и информации с целью удовлетворения: масштабу решаемой задачи с возможностью подключения всех технологических подсистем и общен지니어ного оборудования ускорительного комплекса; предъявляемым СУ требованиям; реализуемости в отведённые сроки. Принятые решения должны инкапсулировать разнообразие управляемого оборудования и быть актуальными в течение длительного времени с учётом развития вычислительных средств, ИТ и возможностью адаптации к ним.

В результате анализа поставленной задачи, с учётом объема потоков данных, их интенсивности и распределения по времени суперцикла, в основу стандартной ОС было положено использование исключительно *Unix*-подобных ОС с акцентом на свободно распространяемую систему *Linux*. На среднем уровне СУ использовалась загружаемая по ТЛВС ОС *LinuxOS*. На верхнем

уровне использовалась *Red Hat Linux 4.1*, однако отсутствие в тот момент на *Alpha* станциях в среде *Linux* удовлетворяющих нашим требованиям графических средств вынудило нас использовать и ОС *DEC Unix*.

Основная концепция разрабатываемого ПО системы управления заключалась в создании единого информационного пространства, отображающего текущее состояние всего ускорительного комплекса и используемого всеми создаваемыми программными компонентами для контроля и управления в РВ. Единое информационное пространство должно обладать, по крайней мере, следующими свойствами:

- *доступность* – любая информационная единица пространства всегда доступна независимо от места её хранения и источника поступления запроса на доступ;
- *безопасность* – надёжная защита всех компонент информационного пространства от несанкционированных внешних воздействий;
- *презентация* – для потребителя всё информационное пространство имеет единую логическую организацию (внешнюю схему) независимо от физической среды хранения данных и распределения данных по вычислительным средствам;
- *адресация* – существует единая система координат всего информационного пространства.

Все программные компоненты СУ разрабатываются и реализуются как множество операций над элементами информационного пространства, обработка данных отделена от интерфейса с человеком и от взаимодействия с оборудованием. На рис. 2 показана принципиальная схема связи основных программно-информационных логических подсистем СУ. В нашей схеме управление объектом эквивалентно редактированию данных в едином информационном пространстве.

В целом, разработанное и созданное ПО верхнего уровня [2] выполняет в *Unix*-подобной среде основные функции, типичные для современных больших систем *SCADA* (*Supervisory Control And Data Acquisition* – диспетчерское управление и сбор данных), каковыми являются:

- обмен данными с контроллерами оборудования (у нас данные в контроллерах оборудования являются частью единого информационного пространства);

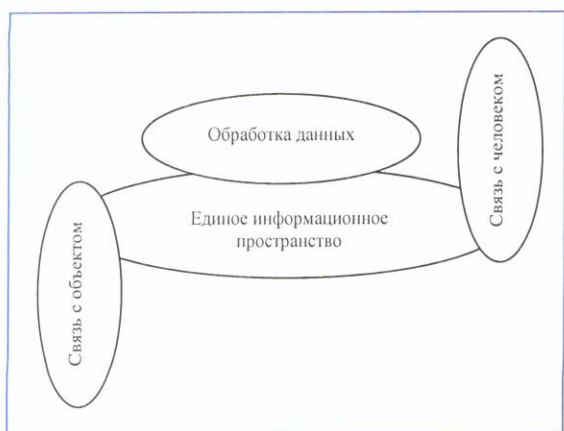


Рис. 2. Базовые структурные элементы ПО системы управления

- распределённая обработка данных в режиме РВ;
- отображение информации на средствах визуализации в физических терминах (у нас они отображаются в табличной и графической форме);
- ведение БД РВ;
- аварийная и предупредительная сигнализация.

Основные программные подсистемы

Управление данными

Для управления в РВ динамическими данными с высоким коэффициентом обновляемости в среде ОС *Linux* на языке *C* была создана Специализированная Система Управления распределёнными Данными (ССУДА) РВ [3]. Её отличительными особенностями являются: трёхмерные таблицы фиксированной размерности; адресация данных по координатам {имя таблицы, номер строки, номер столбца, номер плоскости, номер копии таблицы}; нет поиска по ключам, но возможна циклическая обработка данных по номерам строк, столбцов, плоскостей. Таблицы размещаются в файлах БД, которые распределяются по компьютерам. На одном компьютере может храниться несколько БД. Таблицы, где хранятся аппаратные коды, при необходимости копируются в память контроллеров оборудования сразу после их включения. Распределение БД по компьютерам, а таблиц по БД осуществляет администратор ССУДА, исходя из решаемых задач и оптимизируя время доступа и обработки данных. Прикладной программист не знает о месте нахождения данных, для доступа он использует только их координаты.

Средства управления распределёнными данными включают в себя: библиотеку клиентских функций; набор программ-серверов доступа к таблицам БД на компьютерах, где хранятся БД; системную БД с описаниями топологии распределения БД, таблиц и др. Все взаимодействия осуществляются на базе протоколов *NFS* и *UDP*.

Сейчас в системе управления имеется свыше 2000 таблиц, хранящихся в 12 БД на 8 компьютерах. Из общего числа таблиц около 250 копируются в оперативную память контроллеров оборудования. Совокупность всех таблиц ССУДА, отображающая текущее состояние ускорительного комплекса со всеми его описаниями, образует *единое информационное пространство* с доступом к данным из любого имеющего разрешение компьютера. Распределение данных по дискам серверов и памяти контроллеров оборудования в виде трёхмерных таблиц создаёт основу *единой формы представления* информационного пространства РВ. *Единая система координат* адресации любого элемента информационного пространства {имя таблицы, номер строки, номер столбца, номер плоскости, номер копии таблицы} инкапсулирует и распределение данных, и доступ к КО. Защита от несанкционированных внешних воздействий обеспечивается стандартными средствами ОС *Linux* и специальными возможностями ССУДА.

Обработка данных

Выделяются два основных потока данных по уровням СУ: *первый* – измеренные значения сигналов, перемещаемые снизу вверх; *второй* – управляющие значения параметров, перемещаемые сверху вниз. Основной

объём динамических данных представляют собой измененные значения, которые каждый суперцикл обновляются на 100 %. До 90 % трафика в коммуникационной среде СУ составляет информация, связанная с измеренными данными и рассылкой обновлённых данных по активным консолям. Человек в СУ является составной частью всех контуров регулирования и инициализирует передачу данных второго потока. Поэтому и прикладные программы верхнего уровня СУ в среде *Linux* разделены на две категории: программы управления и измерительные программы-супервизоры. Супервизоры разрабатываются для каждой технологической подсистемы. Они постоянно активны, осуществляют сбор и обработку измеренных данных, проводят мониторинг состояния объекта управления и самой СУ и готовят, в случае необходимости, аварийно-предупредительную информацию.

Управляющие прикладные программы создавались в форме загружаемых выполняемых модулей, каждый из которых осуществляет обработку группы взаимосвязанных параметров. Модуль проверяет на допустимость поступившие от оператора новые значения физических параметров, пересчитывает по соответствующим алгоритмам все связанные с ними значения других физических параметров и записывает результаты в соответствующие таблицы. Полученные физические значения пересчитываются в коды аппаратуры, называемые технологическими параметрами, которые записываются в копии таблиц БД на диске и в памяти контроллеров оборудования.

Разработка новой прикладной программы и проектирование соответствующих данных с учётом возможных режимов работы технологического оборудования выполняются одновременно. По результатам проектирования в определённых БД создаются требуемые таблицы фиксированного формата для хранения значений новых параметров, которые расширяют единое информационное пространство. Все разрабатываемые прикладные программы (и супервизоры, и управляющие) реализуют множество допустимых операций над элементами единого информационного пространства и выполняются как фоновые процессы без связи с терминалами.

Основной задачей прикладных программ в контроллерах оборудования, выполняющихся под управлением специализированной ОС, является синхронизированный с событиями ускорителя обмен информацией между интерфейсной электроникой и расположенными в оперативной памяти КО копиями таблиц ССУДА. Они осуществляют связь единого информационного пространства с объектами управления и обеспечивают соответствие информационного пространства текущему состоянию ускорительного комплекса.

На каждом из компьютеров верхнего уровня, где могут выполняться прикладные программы, постоянно загружена программа “загрузчик”, который по сетевому запросу загружает с файлового сервера прикладной модуль с указанными параметрами. Таким образом, один прикладной модуль одновременно может быть загружен с разными внешними параметрами в один или несколько компьютеров и одновременно обрабатывать разные комплекты данных по одному алгоритму.

GUI - графический пользовательский интерфейс

Графический пользовательский интерфейс был реализован на языке *C* в среде *DEC Unix* на базе графических средств *X Window*, *Motif* и пакета *XRT*. *GUI СУ* – это комплекс программ, полностью управляемых данными (*data driven*), т.е. соответствующими описаниями в БД ССУДА (рис. 3). Он инициализируется на каждой консоли пользователем при открытии сессии взаимодействия с СУ, а сессия пользователя – это навигация в физических и ускорительных терминах и выполнение конкретных действий из множества допустимых операций в едином информационном пространстве СУ. Каждый пользователь имеет определённые права по управлению оборудованием для каждой ускорительной установки. Просмотр данных разрешён всем и всегда в полном объёме, а все консоли СУ равнозначны и идентичны по правам и возможностям независимо от местоположения, что автоматически следует из свойств единого информационного пространства. Специальная программа (выполняющийся в среде *Linux* диалоговый сервер) контролирует открытые сессии взаимодействия с СУ, разрешает или запрещает операции в соответствии с правами пользователя и защищает от несанкционированных

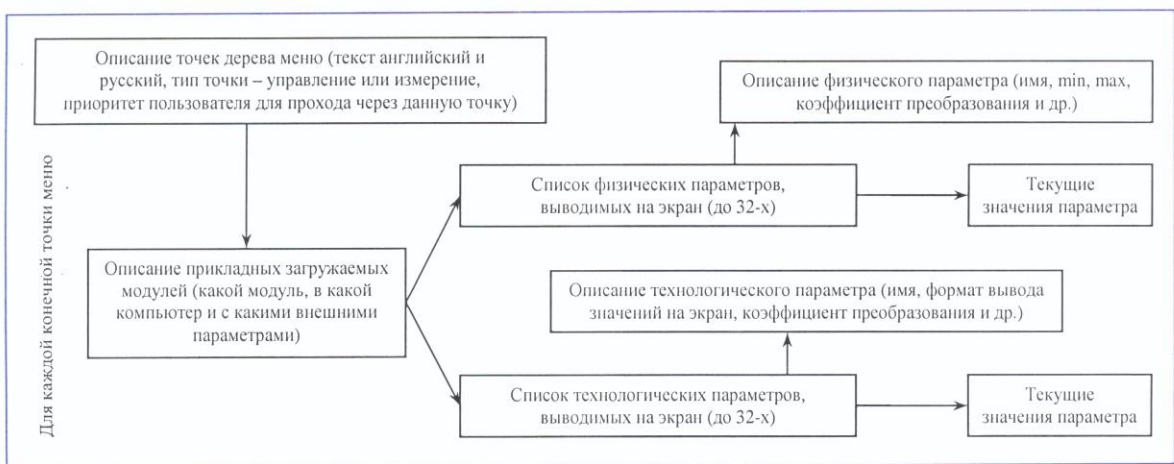


Рис. 3. Основные объекты БД, описывающие пользовательский интерфейс

или коррелирующих воздействий операторов на ускорительный комплекс при их одновременной работе в разных помещениях различных зданий.

Графический пользовательский интерфейс позволяет выбирать в дереве меню (рис. 4) конечную точку – задачу оператора, просматривать и редактировать относящиеся к выбранной задаче данные в табличной форме, наблюдать подготовленные прикладными программами графические представления наиболее важных параметров и функциональных зависимостей. Отображение текущих значений динамических параметров в таблицах и графическое представление на всех активных консолях обновляются каждый суперцикл. В настоящее время дерево меню содержит свыше 3800 точек, из них более 2700 – конечные.

Задачи оператора сгруппированы по технологическим подсистемам и ускорительным установкам. При выборе в меню конкретной конечной точки открывается окно табличного представления текущих значений параметров, описанных в БД и имеющих отношение к данной задаче. Если задача подразумевает управляющие воздействия, то в соответствии с описанием задачи оператора в указанных серверах загружаются указанные прикладные модули, в обязанности которых входит обработка значений параметров, отображаемых в таблице на экране. Свои операции прикладные программы выполняют над комплектом данных, соответствующим выбранному режиму работы ускорителя. Для просмотра значений измеряемых параметров данные готовят супервизоры. Каждый суперцикл в СУ обновляются более 50 файлов-картинок, которые в графической форме отображают наиболее важные параметры работы ускорительного комплекса У-70 и могут выводиться на экраны консолей по запросу операторов.

В целом, графический пользовательский интерфейс исполняет роль “проводника” по единому информационному пространству и “инициализатора” распределенной обработки данных в соответствии с требованиями пользователей. Он осуществляет связь человека с единым

Today: Fri Oct 26 09:22:50 2007

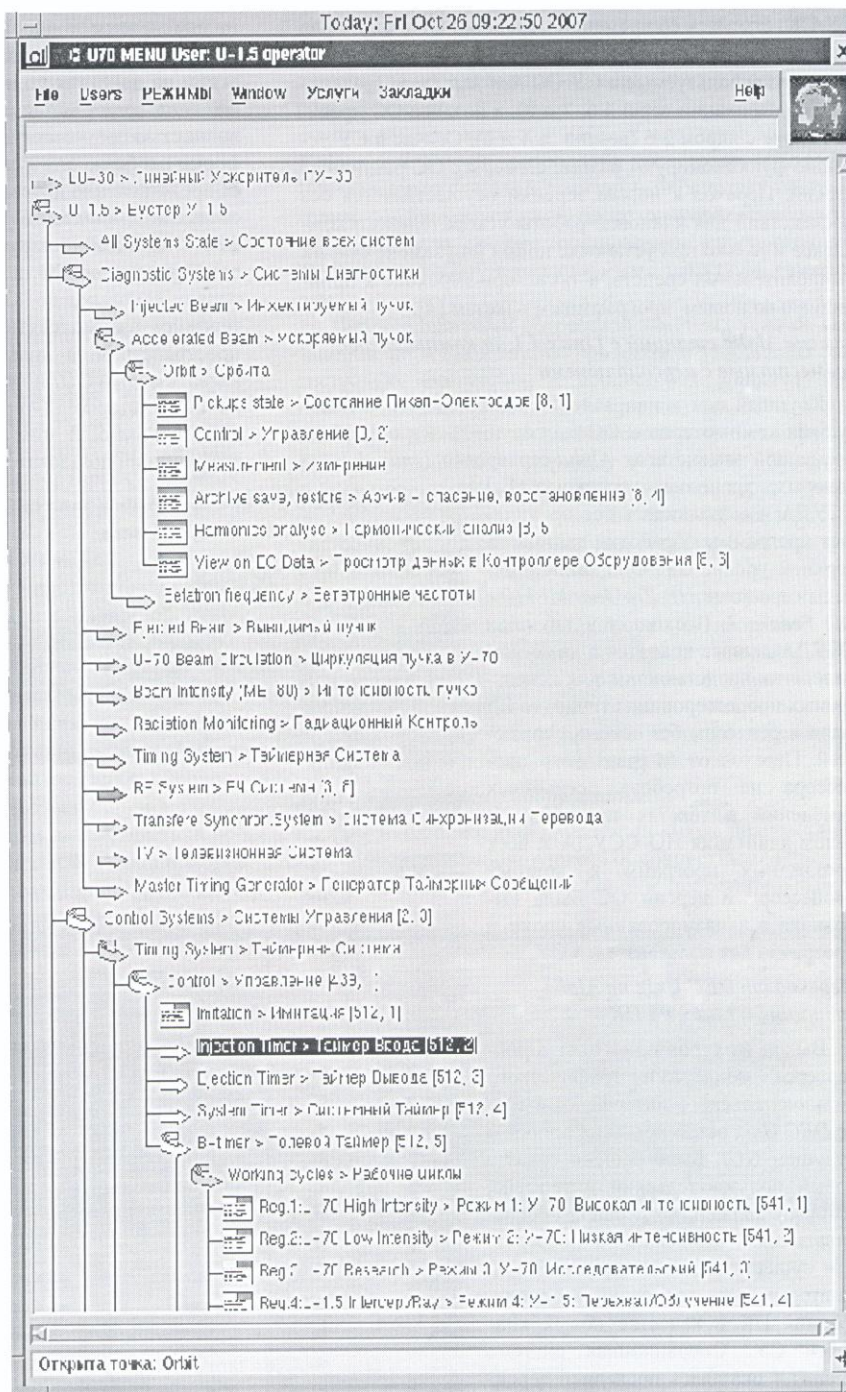


Рис. 4. Часть дерева меню в ОС DEC Unix, взятая с экрана X-терминала

информационным пространством в терминах установок и физических параметров. Все сетевые взаимодействия GUI-программ построены на базе протокола UDP.

Переход к однородным компьютерам с ОС Linux

К 1999 г. характеристики ПК приблизились и постепенно начали превосходить характеристики Alpha станций. Поскольку в наших задачах разрядность процессора не имеет решающего значения, мы стали включать в

СУ ПК, стремясь со временем перевести СУ полностью на совместимые ПК и ОС *Linux*. При этом эволюционировали и используемые в СУ ОС *Red Hat Linux* (различные модификации версий 6, 7 и 9), а позднее ОС проекта *Fedora* с ядром 2.6 (версии 3, 4 и 6). Сейчас в СУ успешно функционируют разные серверы с ОС различных версий. Переход к новым версиям осуществляется без последствий для плановой работы ускорительного комплекса и только при установке новых или замене старых вычислительных средств, а также при переходе к принципиально новым программным пакетам [4].

Замена Alpha станций с Linux 4.1 на компьютеры, совместимые с персональными

Крупный шаг в направлении однородности используемых компьютеров и ОС был осуществлён при одновременной замене всех *Alpha*-серверов с *Linux 4.1*, на которых хранились основные БД ССУДА и выполнялись все прикладные программы обработки данных на верхнем уровне СУ. Они были заменены серверами *HP ProLiant ML110* с ОС *Fedora 3*. Поскольку в таблицах ССУДА данные хранятся в двоичном виде, а их представление для используемых процессоров идентично, то БД были перенесены без особых сложностей. Переход от 64-разрядного процессора не потребовал серьёзных изменений в текстах программ. В целом адаптация ПО ССУДА и всех прикладных программ к другому процессору и версии ОС была выполнена в запланированные сроки и прозрачно для пользователей СУ.

Переход от DEC Unix на Alpha станциях к Linux на ПК

Выход из строя одного из *Alpha*-серверов поддержки графического пользовательского интерфейса в среде *DEC Unix* остро поставил вопрос о будущем *GUI*. Было решено создать новый пользовательский интерфейс, который должен, во-первых, обеспечивать связь человека с СУ посредством существующих *X*-терминалов и, во-вторых, дать возможность использования ПК в качестве новых консолей СУ. Операционная система *Fedora 4* оказалась последней версией, с которой стабильно работал старый *X*-сервер наших *X*-терминалов. Поэтому она использовалась в качестве платформы для создания нового графического пользовательского интерфейса на языке *C* на базе пакета *GTK+*.

Поскольку в основе нового интерфейса лежат те же принципы, а строился он на тех же самых описаниях, хранящихся в таблицах ССУДА,

то изменениям подвергался только программный слой непосредственного управления объектами экрана и реакции на действия пользователя. Поэтому, с точки зрения пользователя, по содержанию новый интерфейс полностью соответствует старому, а отличается лишь по форме отображения и цветовой гамме. Рис. 5 показывает представление дерева меню в новом интерфейсе. При создании и внедрении этого пользовательского интерфейса не вносилось никаких изменений в существующее прикладное ПО, а старый интерфейс используется наряду с новым, т.е. СУ имеет одновременно два интерфейса взаимодействия с единым информационным пространством, а пользователи могут постепенно осваивать новый *GUI* и привыкать к нему. По мере выхода из строя старых *X*-терминалов, они замещаются ПК с ОС *Fedora 6*. В этом случае программы, реализующие графический пользовательский интерфейс, выполняются



Рис. 5. Часть дерева меню в ОС *Linux*, взятая с экрана монитора ПК

непосредственно на ПК, а X-протокол больше не нагружает ТЛВС. После перевода всех консолей на ПК серверы с *DEC Unix* будут сняты с эксплуатации.

Некоторые новые консоли пультов управления по желанию операторов ускорители оснащаются двойными мониторами (*dual monitors*), т.е. два монитора подключаются к одному ПК. Тогда рабочий стол *GNOME* располагается на экранах двух мониторов, подключенных к одной ПК. На рисунках 6 и 7 показано изображение реального рабочего стола на пультах управления, сделанное программой *import*.

Переход от LynxOS и процессора Motorola к Linux на ПК

Под управлением ОС *LynxOS* во всех *VME* работает пакет одних и тех же написанных на языке *C* программ, которые постоянно находятся в оперативной памяти. Программы обеспечивают транспортировку данных между верхним и нижним уровнями СУ, т.е. между ТЛВС и магистралями *MIL 1553*. Со стороны ТЛВС на базе стандартных протоколов *UDP* и *TCP* построены специализированные прикладные протоколы, которые позволяют загружать в память контроллеров оборудования двоичные программы, конфигурационные файлы, копии таблиц ССУДА, которые становятся частью единого информационного пространства, а также осуществлять доступ к содержимому хранящихся в памяти КО копий таблиц ССУДА и поддерживать взаимодействие виртуального терминала с ОС контроллеров оборудования.

При реализации межпрограммных взаимодействий использовались специфические для *LynxOS* функции, которые обеспечивают создание многопоточных процессов

(*multi-threads*), создание и доступ к разделяемой памяти, семафорам и сигналам. Используются две разделяемые памяти для передачи данных сверху вниз и снизу вверх, соответственно.

Для замены каркасов *VME* установлены ПК с ОС *Fedora 6* и платой контроллера магистрали *MIL 1553*. Поставляемые с платой драйвер и библиотека подпрограмм поддерживают свой протокол управления магистралью. Было решено организовать многопоточные процессы и работу с разделяемыми памятьями, семафорами, сигналами в соответствии со стандартом *POSIX*.

Перенос программ потребовал заметных усилий, вызванных не только другим протоколом управления магистралью. Во-первых, в процессорах *Motorola 68K* принят порядок байт от старшего к младшему (*big-endian*). В ПК с *x86*-процессорами принят порядок байт от младшего к старшему (*little-endian*), а обмен с магистралью *MIL 1553* осуществляется через промежуточную 16-разрядную память как в контроллере магистрали, так и в оконечных устройствах. Во-вторых, в использованных нами библиотеках *LynxOS* чтение/запись разделяемой памяти выполняется без кэширования, а поток одного процесса мог напрямую послать сигнал конкретному потоку другого процесса. В варианте *POSIX* работа с разделяемой памятью выполняется с кэшированием, а сигнал можно послать только головному потоку другого процесса, что меняет синхронизацию потоков разных процессов.

Linux и нестандартное оборудование

Для некоторых информационно емких диагностических подсистем используются ПК совместимые машины с операционной системой *Linux* и дополнительными

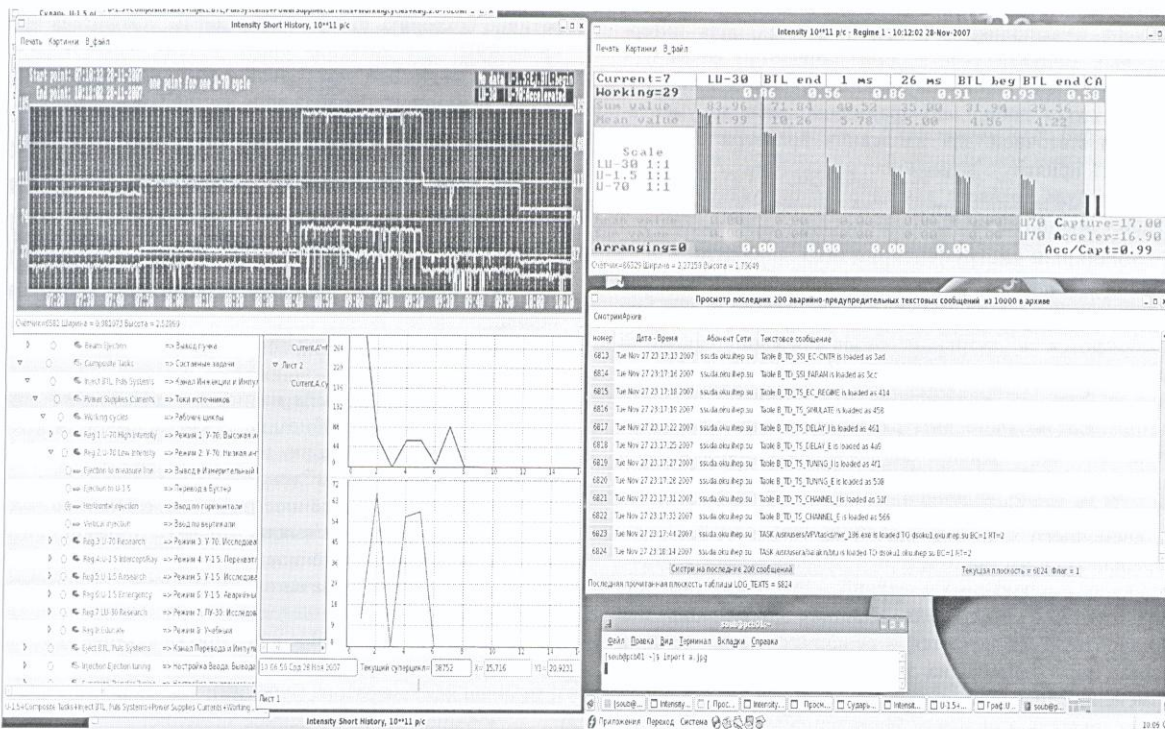


Рис. 6. Рабочий стол двойных мониторов консоли пульта управления бустера

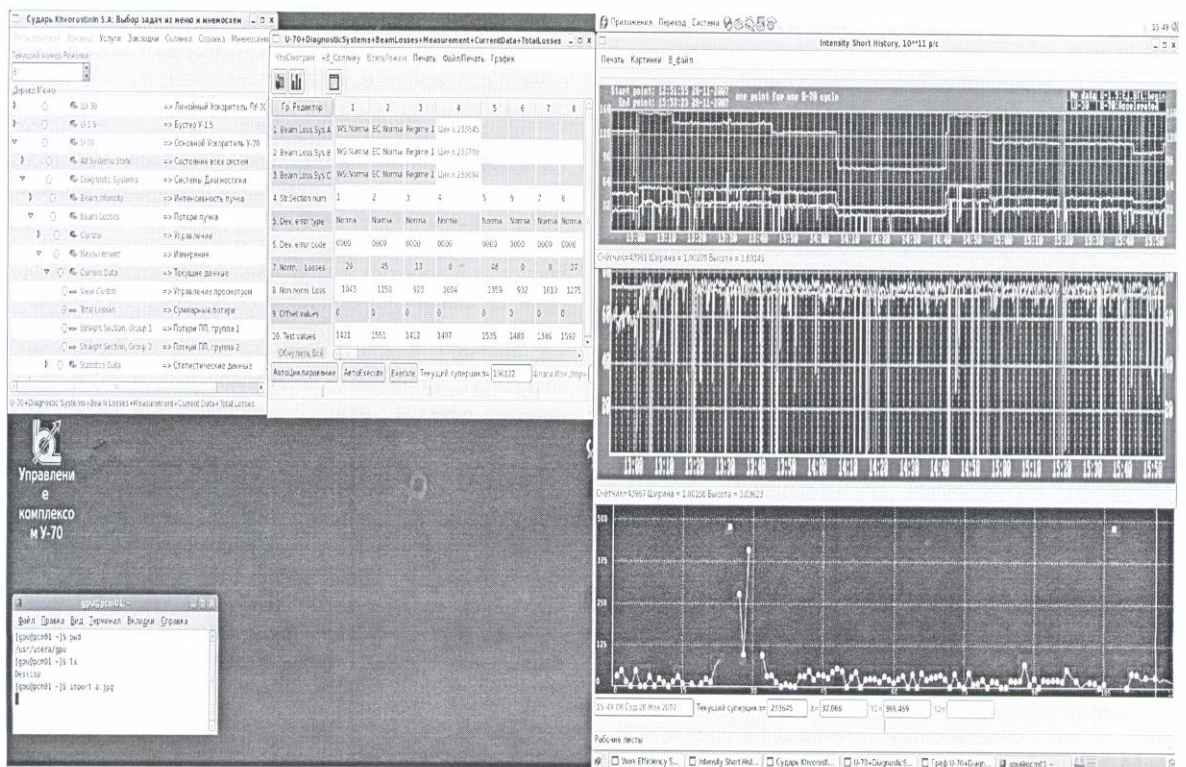


Рис. 7. Рабочий стол двойных мониторов консоли главного пульта управления

интерфейсными платами, установленными непосредственно в компьютер, для связи со специальным оборудованием [5, 6]. В этом случае ПК выполняет функции и верхнего, и нижнего уровня СУ. Он осуществляет связь с оборудованием, содержит, по крайней мере, одну БД ССУДА и выполняет обработку данных. Для интерфейсных модулей собственной разработки создаются драйверы в ОС *Linux*. Закупаемые модули должны сопровождаться либо *Linux*-драйвером, либо документацией, достаточной для написания драйвера. Хотелось отметить приятную возможность в системах *Fedora* собрать модуль драйвера для ядра 2.6 без установки исходных текстов ядра ОС.

Специальный инструментарий исследователя

С целью обеспечения проведения исследовательских работ на ускорительном комплексе, которые для СУ заключаются в активном редактировании значений управляющих параметров и одновременном наблюдении за большим количеством взаимосвязанных параметров, СУ предоставляет ряд специальных возможностей. Для всех управляющих физических параметров в информационном пространстве обязательно резервируется место для комплекта текущих значений и комплекта временного хранения значений. Также можно резервировать в БД место для 30 комплектов длительного хранения именованных значений – это архив интересных режимов ускорителя. Процесс управления для пользователя заключается в редактировании значений параметра в таблице на экране, что реально означает изменение значений в комплекте текущих значений. Исследователь в процессе управления может оперативно копировать данные из одного комплекта в другой

или сравнивать данные любых комплектов. Существует команда актуализации текущих значений, т.е. производится перерасчет физических параметров в технологические коды и запись их значений в таблицы в памяти контроллера оборудования. Следовательно, можно оперативно сохранять интересные режимы ускорителя или промежуточные значения управляющих параметров в процессе настройки ускорителя и восстанавливать их в случае неудачного продолжения настройки.

Графический пользовательский интерфейс при редактировании данных, представленных на экране в табличной форме, поддерживает возможность группового редактирования, состоящего из двух этапов.

1. Выбор в таблице группы редактируемых данных, который можно выполнить одним из трёх способов:

- выбрать все элементы строки или диапазона строк;
- выбрать все элементы столбца или диапазона столбцов;
- выбрать элементы на пересечении строки или диапазона строк со столбцом или диапазоном столбцов.

2. Выполнить операцию над множеством значений выбранных элементов таблицы:

- присвоить введённое значение (например, одной командой включить/выключить группу устройств);
- прибавить введённое значение (например, увеличить на константу значения токов опоры группы источников питания);
- вычесть введённое значение (например, уменьшить на константу значения токов в последовательности узлов для генератора функций);
- умножить на введённое значение;
- разделить на введённое значение.

При проектировании данных для важных измеряемых параметров можно зарезервировать место в таблицах ССУДА для архива измеренных значений в двадцати девяти предыдущих циклах. Пользовательский интерфейс обеспечивает просмотр архива в табличной форме.

Кроме просмотра любых из обновляющихся каждый суперцикл файлов-картинок, подготовленных прикладными программами, пользователь может запросить представления в форме графика или столбчатой диаграммы функциональных зависимостей значений любых физических параметров, хранящихся в едином информационном пространстве, наблюдать дрейф или колебания их значений от цикла к циклу. Для этих целей создаётся окно так называемой стандартной графики. В рабочем поле окна может быть создано до четырёх рабочих листов, на каждый из которых можно выводить до восьми графиков. На каждом листе поддерживается три возможных режима вывода графиков функциональных зависимостей:

1) обновляемый каждый суперцикл обычный график текущих значений;

2) накопление на рабочем листе графиков текущих значений до определённого порога, после чего выводится график текущих значений с графиками минимальных и максимальных значений во всех точках за время слежения;

3) текущие значения параметров берутся за эталон и впоследствии всегда выводятся в виде постоянного графика на рабочий лист вместе с графиком меняющихся во времени текущих значений.

Заключение

Созданная СУ успешно эксплуатируется уже достаточно длительное время. В настоящее время, практически, всё созданное ПО функционирует в среде *Linux*. В сеансе в устоявшемся крэйсерском режиме работы ускорительного комплекса в недрах СУ, как правило, в среднем одновременно открыто десять-двенадцать пользовательских сессий, загружено более 180 прикладных процессов, поддерживающих единое информационное пространство, осуществляющих сбор и обработку данных в режиме РВ и обеспечивающих связь информационного пространства с человеком и объектом управления.

После выключения ускорителей единое информационное пространство в распределённой среде *Linux* отображает их последнее состояние. Компьютеры верхнего уровня СУ никогда не выключаются. Поэтому в любое время в любом помещении любого здания можно включить какое-то оборудование и оно сразу будет под контролем СУ. После включения электропитания контроллера оборудования на верхний уровень СУ посылается сообщение об этом событии и соответствующие таблицы с данными с дисков серверов верхнего уровня СУ копируются в память КО, тем самым расширяя единое информационное пространство на нижний уровень СУ. Затем в контроллер оборудования загружаются описанные для данного контроллера в БД ССУДА прикладные

программы и, при необходимости, конфигурационные файлы. При стартовом запуске прикладные программы в памяти контроллера всегда копируют управляющие данные из таблиц, загруженных в память контроллера, в аппаратуру. Именно так происходит автоматическое восстановление последнего рабочего состояния ускорительного комплекса после подсадки напряжения или после включения ускорителя, а единое информационное пространство всегда соответствует текущему состоянию объекта управления. Таким образом, нижний уровень СУ – это отключаемая/подключаемая часть единого информационного пространства. Отключили – дисковая копия управляющих данных хранится на верхнем уровне СУ. Включили – копия управляющих данных автоматически восстанавливается с диска на нижнем уровне СУ и в ускорительном оборудовании.

К настоящему моменту число контролируемых СУ технологических подсистем ускорительного комплекса достигло тридцати девяти и это не предел. Есть чёткие правила расширения единого информационного пространства. Подключение к СУ новых технологических подсистем по существу означает расширение единого информационного пространства, соответственно, расширяется и связь с объектом управления и взаимодействие с человеком. В настоящее время мы не видим ограничений на возможности наращивания СУ, так как все данные собираются, обрабатываются и отображаются в заданном виде на экранах всех установленных в СУ консолей в каждом суперцикле. Переход к однородной вычислительной и операционной среде придал СУ архитектурно и концептуально завершённый характер.

Работа выполнена в ФГУП ГНЦ "Институт физики высоких энергий" (г. Протвино, Московская обл.).

Контактный телефон (4967) 71-38-03.

E-mail: Valery.Voevodin@ihep.ru

Список литературы

1. Воеводин В.П., Комаров В.В., Миличенко Ю.В., Перриолла Ф. Система управления ускорительным комплексом У-70 // Приборы и системы управления. 1999. № 6.
2. Voevodin V. Software Architecture of the U-70 Accelerator Complex New Control System // Proceedings of the ICALEPCS'99. Italy, Trieste, 1999.
3. Воеводин В.П. ССУДА – Специализированная Система Управления распределёнными Данными реального времени // XVI совещание по ускорителям заряженных частиц, Протвино, 1998. Т. 1.
4. Kim L., Klimenkov E., Komarov V. et al. Experience with the new control system of IHEP accelerators complex // Proceedings of the ICALEPCS'2005. Switzerland, Geneva, 2005.
5. Klimenkov E., Mamuchashvili N., Milichenko Yu., Voevodin V. Distributed analog signal observation system based on PC's with Linux // The Third International Workshop on Personal Computer and Particle Accelerator Controls (PCaPAC-2000). Hamburg, Germany, 2000.
6. Klimenkov E., Losev G., Milichenko Yu., Voevodin V. Beam parameters measurement based on TV methods // XIX Russian Particle Accelerator Conference RUPAC-2004. Dubna, 2004.