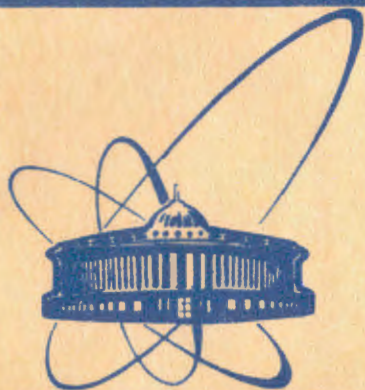


85-200



сообщения
Объединенного
института
ядерных
исследований
дубна

P10-85-200

Г.Балука, И.М.Саламатин

ПЕРСПЕКТИВНЫЕ НАПРАВЛЕНИЯ РАБОТ,
УСКОРЯЮЩИХ СОЗДАНИЕ СИСТЕМ
АВТОМАТИЗАЦИИ ЭКСПЕРИМЕНТОВ

1985

ВВЕДЕНИЕ

Развитие систем автоматизации экспериментальных исследований /САЭ/ идет путем передачи ЭВМ все большего числа функций анализа, планирования, управления экспериментом/1/. Соответственно возрастают трудовые и материальные затраты на создание и эксплуатацию программного обеспечения /ПО/. В связи с этим роль ПО САЭ непрерывно растет и в большинстве случаев затраты на него /в первую очередь, потери времени исследователей и затраты времени разработчиков/ превышают по значимости материальные затраты на технические средства. Специалисты предсказывают/1-3/, что проблема повышения производительности труда программистов будет в числе основных в течение ближайших 10 лет.

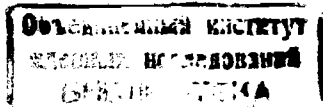
Область интересов авторов - программное обеспечение относительно простых САЭ, количество которых в одной исследовательской организации может достигать нескольких десятков/1,4-6/.

В данной работе содержится обсуждение проблемы "насыщения" исследовательской организации программным обеспечением таких систем, определение требований к ПО САЭ, обзор и анализ средств и методов, используемых при создании ПО САЭ, и формулировка методики динамического формирования конфигурации ПО САЭ. Ниже для краткости на эту проблему будем ссылаться без указания ее названия.

1. ОБСУЖДЕНИЕ ПРОБЛЕМЫ И ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ К ПО САЭ

Современное состояние условий разработки и использования средств автоматизации характеризуется ускоряющимся ростом количества ЭВМ и их доступности, диспропорцией в уровнях развития технологии производства электронных схем и программного обеспечения, углубляющимся дефицитом высококвалифицированных программистов, сложившейся практикой разработки ПО САЭ потребителями САЭ - самими исследовательскими организациями, преимущественной ориентацией на использование программно совместимых ЭВМ /серии СМ, "Электроника-60" и др./ и оборудования КАМАК. Прогнозируется/1/ плавное изменение основных условий, в связи с чем разработки, ориентированные на ближайшие 5-10 лет, могут опираться на их нынешнее состояние.

В работах/1,7/ для решения обсуждаемой проблемы предложен ряд направлений исследований и разработок, практическим выходом ко-



торых должны явиться средства для разработчиков типовых проблемно-ориентированных САЭ и для пользователей-исследователей, работающих на конкретных системах^{1/1/}.

Рассмотрим ряд аспектов обсуждаемой проблемы с целью определения соответствующих требований к ПО систем автоматизации конкретных экспериментов. Здесь, рассматривая требования к ПО САЭ, и далее, при изучении средств и методов создания САЭ, мы в качестве критерия их качественной оценки будем руководствоваться ожидаемым характером увеличения отношения продолжительности периода эксплуатации к продолжительности жизненного цикла ПО САЭ.

1.1. В жизненном цикле программной системы можно выделить три фазы:

- 1/ проектирование;
- 2/ реализацию;
- 3/ эксплуатацию и сопровождение.

Будем считать, что фаза проектирования охватывает все этапы, завершающиеся созданием проекта программной системы, фаза реализации - этапы от начала кодирования и включая ввод ее в эксплуатацию. В сумме этапы этих двух фаз понимаются как разработка программной системы. Термин "создание" используется нами для систем в случае возможности отторжения их от разработчиков и эксплуатации без их участия, либо как синоним слова "разработка" там, где это несущественно.

Известно, что разработка, сопровождение и охватываемые ими этапы являются процессами итерационными. Продолжительность и содержание итераций зависят от многих факторов, из которых наиболее существенными являются выбранный подход к созданию и структура продукта, т.е. используемой системы. Представляется необходимым поиск эффективных решений задач, характерных для каждой фазы жизненного цикла ПО САЭ, с целью сократить продолжительность и число операций в этих итерациях.

1.2. Специфика использования специалистов и оборудования в научных исследованиях такова, что повышение производительности программистов приведет лишь к частичному решению обсуждаемой проблемы. Необходимо выбрать решения, обеспечивающие эффективное в экономическом и профессиональном плане использование специалистов /исследователей, разработчиков САЭ, эксплуатационного персонала/, занятых исследовательской работой либо обслуживающих ее. Поскольку число исследователей обычно существенно превышает число разработчиков ПО, необходимо предусмотреть возможность отторжения ПО САЭ от разработчиков. Необходимыми условиями возможности отторжения являются наличие в ПО САЭ ориентированных на пользователя средств модификации или адаптации ПО к изменяющимся методике эксперимента и условиям работы ПО; методическая обеспеченность систем^{8/и др.}

1.3. "Сочетание большого числа разнообразных функций, которые требуется автоматизировать, существенной эволюционности и необходимости выполнения практически всех функций при оперативном взаимодействии исследователя с системой"^{1/1/} является отличительной чертой САЭ, увеличивающей стоимость решения задачи создания ПО для таких систем по сравнению с другими системами автоматизации.

Ввиду эволюционности процесса исследования и итерационного характера процесса разработки возможность легкой модификации должна рассматриваться как необходимое свойство систем, обслуживающих исследовательский процесс, а не только как свойство средств сопровождения. Если модификация системы с целью отражения изменения в методике эксперимента может выполняться без перепрограммирования управляющей программы системы и без повторения ее трансляции, то мы будем называть это мобильностью системы относительно методики эксперимента по аналогии с использованием этого термина в^{9/}.

Под комплексированностью компонентов САЭ будем понимать возможность переноса этих компонентов в системы для других экспериментов /если требуется аналогичная функция/ без их модификации^{9/}.

Обеспечение указанной мобильности систем и комплексированности компонентов в рамках определенного класса или круга задач может дать наиболее существенное увеличение относительной продолжительности периода эксплуатации ПО САЭ.

1.4. Практика показывает, что ПО САЭ после нескольких этапов развития исчерпывает доступные ресурсы оперативной памяти. Устранить такое препятствие дальнейшему развитию ПО САЭ можно путем перехода к использованию ЭВМ более высокого класса, организации оверлейной структуры и др. Наиболее эффективной, в плане сокращения процесса адаптации ПО САЭ к условиям дефицита оперативной памяти, является изначальная ориентация на использование динамического распределения памяти для компонентов ПО САЭ.

На основе этого обсуждения сформулированы следующие требования к ПО САЭ:

1. Обеспечение скоростных характеристик /время реакции на внешние события/, надежности, функциональных возможностей, соответствующих требованиям обслуживаемого круга задач.
2. Возможность отторжения системы от разработчиков.
3. Мобильность ПО САЭ по отношению к изменениям методики в рамках определенного круга задач и комплексированность компонентов ПО САЭ.
4. Мобильность ПО САЭ по отношению к изменениям коммутации и способа размещения в каркасах блоков экспериментального оборудования КАМАК.
5. Ориентация на пользователя средств управления системой, средств адаптации ее в изменяющихся условиях эксплуатации и средств развития.

Данные требования акцентируют внимание на свойствах системных компонентов проблемно-ориентированных САЭ.

2. ОБЗОР СРЕДСТВ И МЕТОДОВ СОЗДАНИЯ ПО САЭ

К настоящему времени накоплен обширный опыт разработки и применения систем автоматизации экспериментов на основе использования ЭВМ. Для рассмотрения средств и методов создания ПО САЭ, ориентированных на фазы проектирования, реализации и эксплуатации, мы выбрали языки спецификации САЭ, языки реального времени и проблемно-ориентированные модульные системы.

2.1. Применение языков спецификации САЭ. Понимание важности проблем, связанных с проектированием программного обеспечения, возникло в 60-е годы, когда неалгоритмические аспекты процесса создания системы стали поглощать основную часть затрат на ее проектирование и реализацию. Опыт последних лет показал, что трудности часто возникают из-за использования заказчиками и разработчиками ПО САЭ разных систем понятий. Отсутствие взаимопонимания на стадии определения требований и проектирования компенсируется на последующих стадиях создания САЭ ценой больших /иногда десятикратных/10,11/ потерь.

Известны/12/ десятки различных методов определения требований и проектирования больших программных систем. Наилучшим образом "понятийный барьер" может быть преодолен в подходе, основанном на использовании формального языка спецификации программного /возможно, и аппаратного/ обеспечения системы/13-16/, ориентированного на пользователя. Обычно на таком языке выполняется спецификация всех функциональных компонентов прикладного процесса, иногда - структура управляющей программы. При этом специфицируется функция прикладной операции, а не способ /алгоритм/ достижения результата.

Эффективность данного подхода дополнительно обеспечивается тем, что спецификация системы на формальном языке допускает использование моделирующих программ, а также синтаксический и семантический анализ описания. Некоторые ошибки и различия в трактовке требований заказчиком и исполнителем могут быть идентифицированы на самой ранней стадии работы. Это существенно увеличивает вероятность получения проекта и системы, соответствующих требованиям заказчика. Помимо этого описание системы в виде ее формальной спецификации, понятное всем участникам работы, возникает в самом начале работы, что улучшает понимание задачи.

Важнейшими для задач реального времени особенностями языка спецификации систем являются следующие:

- 1/ спецификация действий и их синхронизации выполняется раздельно;

- 2/ спецификация действий может служить заданием на программирование на процедурном языке;
- 3/ координация действий описывается на не процедурном языке, понятном пользователям, например, в терминах "условие-действие"/17/ с использованием условий, зависящих от состояния системы;
- 4/ спецификация системы на формальном языке доступна для машинной обработки.

Возможная сфера приложения такого подхода очень широка: ядерно-физические исследования/17,18/, медико-биологические/19-21/, баллистика/22/, сети связи, управление. Более детальную его характеристику и библиографию можно найти в работах/15,16,23,24/.

2.2. Использование языков реального времени. Независимо от способов определения требований и проектирования программного обеспечения системы для реализации ее могут использоваться алгоритмические языки, допускающие кодирование обработки внешних запросов прерывания и имеющие средства координации процессов. Языки программирования, обладающие этими двумя свойствами, называются языками реального времени /РВ/. Используются три метода в этом подходе: применение готовых трансляторов с добавлением специальных макросредств или подпрограмм к библиотеке; модификация существующих трансляторов с целью включения дополнительных возможностей; создание новых языков и трансляторов. Первый метод наиболее простой и поэтому чаще всего применяется, однако отмечается/24/ возрастание интереса к использованию второго и третьего методов для преодоления других проблем в разработке программного обеспечения.

При наличии жестких требований к скоростным характеристикам частично или полностью система реального времени программируется на языке ассемблера. По мере распространения трансляторов с языков РВ все шире используются языки РВ высокого уровня. Критические по времени участки результирующей программы обычно реассемблируются и заменяются сегментами, написанными на языке ассемблера.

В соответствии с первым методом в работах/25,26/ предложены стандарты Фортран РВ, Бейзик РВ, включающие специальные операции над задачами: запуск, остановку, циклическое исполнение, привязку операций к определенным моментам времени, дополнительные алгоритмические возможности /например, манипулирование рядами машинных слов/, и операции с оборудованием КАМАК. Определенную подготовку к созданию международных стандартов языков можно усмотреть в национальных стандартах: Coral-66/27/, RTL-2/28/, Pearl/29/, Ада/30/. В 1979 г. имел место следующий уровень распространения этих языков: Coral - от 4000 до 6000 программистов в более чем 100 организациях; Ада - более 10 тыс. программистов; Pearl - около 130 систем, реализованных в ФРГ, язык распространен также в Швейцарии, Англии, СССР/24/.

В настоящее время внимание привлекают новые языки РВ /Modula-2/31/, Ада/30/ и др./, обеспечивающие модульную технику конструирования программного обеспечения. Наиболее важными особенностями этих языков являются структуры данных/32/, возможность структурного программирования, специальные средства модульной организации программ/31/ и др. Отмечаются трудности в использовании средств модульной организации программ/33/ наряду с признанием их необходимости.

2.3. Применение проблемно-ориентированных модульных систем. Проблемно-ориентированные модульные системы /ПОМС/, или развитые пакеты прикладных программ, включают развитый язык заданий, функциональное наполнение и системное наполнение/34-37/.

Язык заданий обычно включает непроцедурные средства, обеспечивающие возможность общения с проблемно-ориентированной системой специалистом, не знакомым с программированием. Язык заданий в сочетании с соответствующей поддержкой, обеспечиваемой системным наполнением, может приносить в проблемно-ориентированную систему свойства, присущие языкам спецификации САЭ /см. раздел 1.1/.

Функциональное наполнение определяет область применимости системы и отражает уровень освоения проблемной области специалистами, занятыми работами по автоматизации. Расширение функционального наполнения проблемно-ориентированных систем, как правило, может выполняться пользователями с помощью штатных систем программирования и поддерживается специальными средствами.

Состав и свойства компонентов системного наполнения определяют набор свойств и степень /полноту/ обеспечения требований, предъявляемых к проблемно-ориентированной системе. Часто встречающимися компонентами системного наполнения ПОМС являются средства, обслуживающие генерацию программ, развитие системы, интерактивное взаимодействие с пользователем, организацию выполнения работы и другие мониторные функции/36/.

Характерной тенденцией/36/ развития этого направления в последние годы является переход к созданию ПОМС с помощью инструментально-базовых систем или метасистем, при котором часть средств базовой системы используется в процессе эксплуатации сформированной ПОМС/17,20/.

Кодирование функциональных и системных компонентов ПОМС может выполняться с помощью систем программирования общего назначения. Известны реализации, допускающие создание модулей функционального наполнения на различных языках программирования/17,20/, однако для получения такой возможности требуются дополнительные специальные разработки/38/.

В настоящее время отсутствуют типовые решения архитектуры /внешних функциональных возможностей/ системного наполнения проблемно-ориентированных систем/1,36/. Работы в этом направлении актуальны/1/ и ведутся в ряде организаций. Подробная

библиография может быть найдена в работах/34-37/. Исследования и реализации проблемно-ориентированных систем автоматизации экспериментов описаны, например, в/17-20,39,40/.

Примерами проблемно-ориентированных модульных систем автоматизации экспериментов могут служить MULTI/39/, Q/40/, САНПО/20/ и др. Рассмотрим коротко систему MULTI/39/.

Система MULTI включает подсистемы регистрации и обработки данных (DA) и управления экспериментом (MULTI). Имеются реализации для двух ОС фирмы DEC: RT-11 и RSX-11M. Все четыре модуля подсистемы MULTI практически идентичны в обеих ОС, модули DA (RTDA и RSXDA) существенно зависят от ОС и ЭВМ. Подсистема MULTI написана в виде, пригодном для использования во многих экспериментах. Регистрируемые события описываются таблицами, что позволяет адаптировать DA к методике конкретного эксперимента. Реализация RT-MULTI позволяет построить систему для конкретного эксперимента в течение двух часов для практически любого набора блоков КАМАК из числа используемых в организации, не выполняя дополнительного программирования/39/. Система RSX-MULTI требует редактирования и трансляции некоторых файлов на языке ассемблера /МАКРО-11/ при изменении конфигурации оборудования КАМАК. Система MULTI используется в 20 различных экспериментах FNAL /Батавия/ и нескольких других организациях/39/.

Однако системы MULTI, Q и некоторые другие сильно зависят от особенностей используемого экспериментального оборудования и не распространяются в составе штатного программного обеспечения ЭВМ, что затрудняет /либо исключает/ возможность использования их в других организациях.

2.4. Обсуждение. Языки спецификации являются полезным средством безотносительно к выбранной технологии программирования САЭ. Затраты на разработку языка могут окупиться, если имеет место разделение специалистов на "заказчиков" и "разработчиков" и тиражирование систем.

Быстрое и успешное развитие языков РВ в последние 10 лет стимулировалось, в первую очередь, нуждами разработчиков автоматизированных систем управления технологическими процессами /АСУТП/. Экспериментальные исследования отличаются от промышленных задач существенно большей динамикой требований к САЭ, и в этом случае отрицательно сказывается отсутствие решений некоторых ключевых для систем РВ вопросов, например: нет единого мнения относительно способа разделения функций, необходимых для реализации САЭ, между операционными системами и языками РВ/24/; нет также и стандартов, регламентирующих способ реализации таких функций в ОС или языках РВ. В результате программы на разных языках РВ оказываются несовместимыми из-за различия семантики операций управления процессами/41/. Следствиями этого являются тенденция построения системы на одном языке программирования и потеря прием-

ственности программ при изменении языка. В подавляющем большинстве случаев программирование достаточно простых САЭ выполняется средствами языков РВ. При этом разработка системы осуществляется независимо и ее доведение занимает 1, чаще 2-3 года/1/.

Создание типовых проблемно-ориентированных модульных систем - подход, близкий к искомому решению обсуждаемой проблемы, если обеспечены свойства мобильности таких систем по отношению к изменениям в определенных пределах конфигурации экспериментального оборудования и методики эксперимента, наиболее подверженных изменению в процессе исследования. В связи с этим ниже предлагается методика динамического формирования конфигурации программного обеспечения систем автоматизации экспериментов, обеспечивающая требуемую мобильность.

3. МЕТОДИКА ДИНАМИЧЕСКОГО ФОРМИРОВАНИЯ КОНФИГУРАЦИИ ПРОГРАММНЫХ САЭ

Методика динамического формирования конфигурации /структуры и связей между компонентами/ программного обеспечения САЭ включает несколько частных методов:

1. Представление структуры управляющей программы САЭ в виде иерархии виртуальных программных процессоров.
2. Динамическое формирование информационной базы САЭ, включающей базу экспериментальных данных и базу информационных моделей процессов, являющейся областью коммуникации процессов.
3. Динамическое формирование состава и связей функциональных компонентов /программных модулей/ САЭ, необходимых для обеспечения работы САЭ на любой конкретной фазе эксперимента.
4. Модульная организация программного обеспечения КАМАК и динамическая настройка таких программ в соответствии с конкретным способом размещения и коммутации блоков экспериментального оборудования/в настоящее время - блоков КАМАК/.

Как показал опыт применения, эти принципы обеспечивают возможность получения свойств САЭ, перечисленных в разделе 2, и продолжить развитие САЭ в целом путем "пошаговой детализации"/42/ параллельно с эксплуатацией. Помимо этого в случае возникновения оснований сохраняется возможность дополнить такую систему инструментальными средствами /язык спецификации, компилятор, средства генерации и др./, обеспечивающими существенное расширение или изменение проблемной области.

Элементы предложенной методики исследовались в течение ряда лет/43-47/ и целесообразность их использования подтверждена практикой /например, /19-21, 49, 50//. Данная концепция является достаточно общей и может быть применена для различных ЭВМ и ОС, а также в задачах АСУТП.

Реализация методики ДФК //46-48/ и др./ выполняется для ЭВМ с системой команд PDP-11 на языке Паскаль в ОС типа RT-11.

В заключение авторы благодарят И.М.Калиниченко и Ф.Я.Дзержинского за полезные обсуждения.

ЛИТЕРАТУРА

1. Велихов Е.П., Выставкин А.Н. УСИМ, 1984, №4, с.3.
2. 1983 Technology Forecast: Special Report: Software Electronic Design, 1983, vol.31, No.1, p.242.
3. Electronic Design, 1983, vol.31, No.1, p.147.
4. Вагов В.А. и др. ОИЯИ, 10-82-351, Дубна, 1982.
5. Рябов Ю.Ф. В кн.: Труды 10 Всесоюзной школы по автоматизации научных исследований. ЛИЯФ АН СССР, Л., 1977, с.23.
6. Бейлин М.В. и др. Препринт ИЯФ СО АН СССР, №82-72, Новосибирск, 1982.
7. Нестерихин Ю.Е. и др. Автометрия, 1974, №4, с.3.
8. Дзержинский Ф.Я. В кн.: Современные методы и средства программирования. Материалы семинара. МДНТП, М., 1981, с.135.
9. Encyclopedia of Computer Science and Engineering. (Ed. by A.Ralston, E.D.Reilly Jr.). Van Nostrand Reinhold Co., N.Y., 1983.
10. Бозм Б. Характеристики качества программного обеспечения. "Мир", М., 1981.
11. Куликов А.Ф. УСИМ, 1984, №4, с.12.
12. Ludwig I., Streng W. EWICS, 1978, TC7, No.149.
13. Ludwig I., Streng W. KFK 2506. Kernforschungszentrum, Karlsruhe, 1978.
14. Жук В.И., Шитиков Б.И. Автометрия, 1976, №1, с.79.
15. Агафонов В.Н. В кн.: Данные в языках программирования. "Мир", М., 1982, с.265.
16. Агафонов В.Н. В кн.: Требования и спецификации в разработке программ. "Мир", М., 1984, с.285.
17. Балука Г. и др. ОИЯИ, P10-12960, Дубна, 1980.
18. Crowley-Milling M.C., Human J.T., Shering G.C. CERN Preprint, 10-CO/74-2, Geneva, 1974.
19. Саламатин И.М., Штарк М.Б., Яновский Г.Я. Автометрия, 1981, №4, с.60.
20. Астафьев С.В. и др. Автометрия, 1984, №4, с.44.
21. Кудряшов В.К., Саламатин И.М., Шестимиров В.М. В кн.: Структура, технологические средства и организация систем автоматизации научных исследований. /Под ред. В.И.Кадашевича/. ЛИЯФ, Л., 1982, с.65.
22. Teichroew D., Hershey E.A. IEEE Trans. on Software Eng., 1977, vol.SE-1, No.1.

23. Goldsack S., Hause V., Halling H. Proc. of the First Eur.Symp. West Berlin, Oct.1979. (Ed. by H.Meyer). NHPC, Netherland, 1980, p.525.
24. Pyle J.C. Proc. of the First Eur.Symp. West Berlin, Oct.1979. (Ed. by H.Meyer). NHPC, Netherlands, 1980, p.209.
25. Heller G. EWICS, TC1, Aug.1978.
26. CAMAC. Real-Time BASIC for CAMAC ESONE Committee. RTB/02, 1976.
27. Webb J. Basic CORAL-66, Prentice Hall, 1976.
28. RTL/2. Language Specification. London WIP IHF, 1974.
29. Brandes J. et al. Elektronische Daten verarbeitung, 1970, No.10, p.429.
30. Язык программирования Ада. "Финансы и статистика", М., 1982.
31. Wirth N. Byte, 1984, No.8, vol.9, p.145.
32. Wirth N. Acta Informatica, 1971, 1, p.1.
33. Gutkhnacht J. Byte, 1984, No.8, vol.9, p.157.
34. Ершов А.П., Ильин В.П. В кн.: Пакеты прикладных программ: проблемы и перспективы. "Наука", М., 1982, с.4.
35. Тамм Б.Г., Тыгу Э.Х. Кибернетика, 1975, №4, с.76.
36. Корягин Д.А., Мартынюк В.В. В кн.: Пакеты прикладных программ: проблемы и перспективы. "Наука", М., 1982, с.124.
37. Бежанова М.М. Пакеты прикладных программ. Учебное пособие. Изд-во НГУ, Новосибирск, 1983.
38. Саламатин И.М., Яновский Г.Я. ОИЯИ, P10-12971, Дубна, 1979.
39. Bartlett J.F. et al. IEEE Trans. on Nucl.Sci., 1978, vol.NS-26, No.4, p.4427.
40. Perry D.G. IEEE Trans. on Nucl.Sci., 1979, vol.NS-26, No.4, p.4494.
41. Williamson R., Horowitz E. Soft.Pract.Exp., 1984, vol.14, No.2, p.135.
42. Wirth N. Comm. of the ACM, 1983, vol.26, No.1, p.70.
43. Дади Л. и др. Программирование, 1978, №2, с.26.
44. Балука Г., Островной А.И. ОИЯИ, P10-13004, Дубна, 1980.
45. Балука Г. ОИЯИ, 10-83-761, Дубна, 1983.
46. Балука Г. ОИЯИ, 10-84-463, Дубна, 1984.
47. Балука Г. ОИЯИ, 10-84-678, Дубна, 1984.
48. Балука Г., Саламатин И.М. ОИЯИ, 10-84-573, Дубна, 1984.
49. Вагов В.А. и др. В кн.: XI Международный симпозиум по ядерной электронике. ОИЯИ, D13-84-53, Дубна, 1984.
50. Гриднев Г.Ф., Саламатина Т.С. ОИЯИ, 10-83-598, Дубна, 1983.

СООБЩЕНИЯ, КРАТКИЕ СООБЩЕНИЯ, ПРЕПРИНТЫ И СБОРНИКИ ТРУДОВ КОНФЕРЕНЦИЙ, ИЗДАВАЕМЫЕ ОБЪЕДИНЕННЫМ ИНСТИТУТОМ ЯДЕРНЫХ ИССЛЕДОВАНИЙ, ЯВЛЯЮТСЯ ОФИЦИАЛЬНЫМИ ПУБЛИКАЦИЯМИ.

Ссылки на СООБЩЕНИЯ и ПРЕПРИНТЫ ОИЯИ должны содержать следующие элементы:

- фамилии и инициалы авторов,
- сокращенное название Института /ОИЯИ/ и индекс публикации,
- место издания /Дубна/,
- год издания,
- номер страницы /при необходимости/.

Пример:

1. Первушин В.Н. и др. ОИЯИ, P2-84-649, Дубна, 1984.

Ссылки на конкретную СТАТЬЮ, помещенную в сборнике, должны содержать:

- фамилии и инициалы авторов,
- заглавие сборника, перед которым приводятся сокращенные слова: "В кн."
- сокращенное название Института /ОИЯИ/ и индекс издания,
- место издания /Дубна/,
- год издания,
- номер страницы.

Пример:

Колпаков И.Ф. В кн. XI Международный симпозиум по ядерной электронике, ОИЯИ, D13-84-53, Дубна, 1984, с.26.

Савин И.А., Смирнов Г.И. В сб. "Краткие сообщения ОИЯИ", № 2-84, Дубна, 1984, с.3.

Рукопись поступила в издательский отдел
19 марта 1985 года.

Принимается подписка на препринты и сообщения Объединенного института ядерных исследований.

Установлена следующая стоимость подписки на 12 месяцев на издания ОИЯИ, включая пересылку, по отдельным тематическим категориям:

ИНДЕКС	ТЕМАТИКА	Цена подписки на год
1.	Экспериментальная физика высоких энергий	10 р. 80 коп.
2.	Теоретическая физика высоких энергий	17 р. 80 коп.
3.	Экспериментальная нейтронная физика	4 р. 80 коп.
4.	Теоретическая физика низких энергий	8 р. 80 коп.
5.	Математика	4 р. 80 коп.
6.	Ядерная спектроскопия и радиохимия	4 р. 80 коп.
7.	Физика тяжелых ионов	2 р. 85 коп.
8.	Криогеника	2 р. 85 коп.
9.	Ускорители	7 р. 80 коп.
10.	Автоматизация обработки экспериментальных данных	7 р. 80 коп.
11.	Вычислительная математика и техника	6 р. 80 коп.
12.	Химия	1 р. 70 коп.
13.	Техника физического эксперимента	8 р. 80 коп.
14.	Исследования твердых тел и жидкостей ядерными методами	1 р. 70 коп.
15.	Экспериментальная физика ядерных реакций при низких энергиях	1 р. 50 коп.
16.	Дозиметрия и физика защиты	1 р. 90 коп.
17.	Теория конденсированного состояния	6 р. 80 коп.
18.	Использование результатов и методов фундаментальных физических исследований в смежных областях науки и техники	2 р. 35 коп.
19.	Биофизика	1 р. 20 коп.

Подписка может быть оформлена с любого месяца текущего года.

По всем вопросам оформления подписки следует обращаться в издательский отдел ОИЯИ по адресу: 101000 Москва, Главпочтамт, п/я 79.

Балука Г., Саламатин И.М.
Перспективные направления работ,
ускоряющих создание систем автоматизации экспериментов

P10-85-200

Обсуждается проблема повышения эффективности программирования при разработке систем автоматизации экспериментов /САЭ/. Рассмотрено применение языков спецификации, языков реального времени, систем модульного программирования. Предложена методика динамического формирования конфигурации /ДФК/ программного обеспечения САЭ. Методика ДФК обеспечивает необходимую гибкость САЭ по отношению к изменениям методики эксперимента на этапе эксплуатации САЭ. Предложенная методика не зависит от типа ЭВМ и операционной системы. Реализация выполняется для ЭВМ типа СМ и ОС RT-11.

Работа выполнена в Лаборатории нейтронной физики ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна 1985

Перевод О.С.Виноградовой

Baluka G., Salamatin I.M.
Promising Approaches of Creating Data
Acquisition Systems

P10-85-200

Some problems of increasing the effectivity of creating and developing data acquisition systems are discussed. The work contains characteristics of the standard approaches and their applications. Authors have focused on specification languages, real time languages and modular programming systems, and have presented a new approach - the method of dynamic creation of program configuration. This new method demonstrates the needed flexibility of the data acquisition system at deviation of the measurement's method on the whole system's life cycle. Although the realization has been performed for RT-11 operation system on the SM microcomputer, it can be easily adopted for other systems and computers.

The investigation has been performed at the Laboratory of Neutron Physics, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna 1985