

# Технология сборки модулей (1975-1989) и сборки информационных, операционных технических и интеллектуальных ресурсов предметных областей знаний в среде E-Sciences Интернет

Е.М. Лаврищева<sup>1,2</sup>

<sup>1</sup>*Институт системного программирования им. Иванникова РАН*

<sup>2</sup>*Московский физико-технический институт*

[lavr@ispras.ru](mailto:lavr@ispras.ru), [lavrishcheva.em@mipt.ru](mailto:lavrishcheva.em@mipt.ru)

**Аннотация.** Приводится описание технологии сборки модулей в ЯП средствами системы АПРОП, сделанной по ГОСДоговору с НиЦЕВТ (1975-1976) под руководством академика В.М.Глушкова для ОС ЕС ЭВМ. АПРОП сдана в ГОСФонд. В 1977г. Применялась АПРОП на ВПК (МНИИПА, Липаев В.В.) по договору с 1978-1985г. для реализации программных комплексов Прометей, Яуза, Руза и бортовых приборов для авиации, космоса и флота. Отмечается широкое применение метода сборочного программирования модулей в нашей стране и за рубежом. Определен 2 вариант метода сборки информационных, технических, компьютерных, операционных и интеллектуальных ресурсов в классе предметных областей знаний (физики, математики, медицины, биологии) с использованием средств E-SCIENCES (GRID, ETICS, Семантик Веб. Приводятся лекции и доклады про сборку систем и их семейств СПС из объектов, компонентов, сервисов, geuses в среде . WWW3C. Показаны новые книги, учебники по аспектам технологии сборки 2 и примеры реализации предметных областей знаний с участием специалистов МФТИ и ИСП.

**Ключевые слова:** ресурс, сборка, компонент, сервис, система, брокер, конфигурационная сборка, безопасность, надежность, качество, защита данных.

## Введение

В период 1975-1976гг. в НиЦЕВТ (Райков Л.Д.) была создана система АПРОП для автоматизации и сборки модулей. Система АПРОП была принята в эксплуатацию в ОС ЕС ЭВМ Госкоммисией и передана для использования в ГОСФонд СССР (1977г.). В рамках ВПК (1977-1985) использовалась система АПРОП при создании методом сборки сложные программно-технические комплексы Прометей, Яуза, Руза из модулей в ЯП операциями CALL, link и интерфейсных посредников связи (stub, skeleton) для обмена данными и преобразования неэквивалентных типов данных согласно стандарта ISO/IEC 12207: GDT 1982 к формату ЭВМ с помощью Библиотеки примитивных функций (БМИ), реализованной для сред ЭВМ – ОС ЕС, IBM-360, Corba, MS.Net, Oberon, Unix, Intel, Linux (1978-1985). Библиотека реализована в книге Лаврищевой Е.М., Грищенко В.Н. «Связь разноразличных модулей в ОС ЕС ЭВМ под руководством академика В.М. Глушкова.. В рамках ВПК СССР работы проводились в МНИИПА под руководством доктора тех. наук Липаева В.В. с использованием системы АПРОП и библиотеки БМИ (1978-1984) созданы комплексы Прометей, Протва, Яуза, Руза и бортовые средства (более 100) для авиации, морфлота и космоса. (см. книга

Липаева В.В. «Фрагменты истории развития отечественного программирования для специализированных ЭВМ в 50-80-е годы. –Синтег. Москва -2003.

В СССР сформировалось отечественное сборочное программирование прикладных и технических систем в классе авиационных, космических, морских систем, систем управления, экономики, строительстве мостов, домов и т.п. Основные разработчики технических, программных комплексов ВПК и бортовых устройств награждены премией СМ СССР в 1985г. в день космонавтики. В 1987г. была выдана награда ГКНТ СССР на Европейской конференции «Интерфейс СЭВ- 87» разработчикам Лаврищевой Е.М. и др. ВПК программные средства переданы в ЕрНУЦ (1985) с функциями преобразования данных БМИ в системных средах: Sun Microsystems, .Net, VS.MS, Corba, Oberon, IBM, Intel, Unix и др. [1–5]. На основе материалов по отечественному методу сборки в СССР и за рубежом комитет IEEE создал стандарт 828: Configuration, 1998 для конфигурационной сборки структуры систем операциями config, building, integration, assembly.

## 1. Пути развития работ по сборочному программированию в 1960годах

Наряду с АПРОП Глушкова В.М. разрабатывались средства автоматизации программ Альфа и Вета А.П. Ершовым в СО РАН СССР, МГУ, ЛГУ и др. А.П.Ершов в 1984году на конференции в Москве сказал «сборочное программирование обеспечивает построение уже существующих (проверенных на правильность готовых отдельных фрагментов КПИ, модулей (типа reuses) в сложную программную структуру» и составляет основу программирования в стране, наряду с синтезирующим Э.Х. Тыгу и конкретизирующим программированием»...А.П. Ершов предложил стать оппонентом докторской диссертации Лаврищевой Е.М.«Методы, средства и инструменты сборочного программирования при представлении реферата диссертации под рук. академика В.М.Глушкова [6]. Но поскольку В.М.Глушков умер на 59 году жизни 31.01.1981г. потребовалось представить для защиты текст диссертации и собрать отзывы с организаций Прибалтики (Эстония, Латвия, Литва), Узбекистана, Молдавии и Грузии. Защита состоялась в 1988 году (ВАК СССР).

Основу сборки модулей составлял интерфейсный посредник (stub, skeleton), в котором описывались операции связи (Call, link) и преобразование неэвивалентных данных функциями БМИ. По сборочному программированию была защищена диссертация – Лаврищевой Е.М. Методы, средства и инструменты сборочного программирования. -1988. -37с. (ВАК СССР) и диссертация 347с. с отзывами на диссертацию в базовых городах –Москва, Ленинград, Таллин, Рига, Вильнюс, Новосибирск, Кишинев, Тбилиси, Кишинев (ВАК СССР, 1989).

Написаны книги по сборочному программированию участниками работ в ВПК:

\* - Лаврищева Е.М, Грищенко В., Н. Сборочное программирование. -М.1991.-137с.

- Липаев В.В., Позин Б.А., Штрик А.А. Технология сборочного программирования.-М.-1992.-208с.

-Лаврищева Е.М, Грищенко В., Н. Сборочное программирование. Основы индустрии программных продуктов. К.: Наук. Думка. - 2009.-371с.

Эти книги широко использовались на отечественных ЭВМ в 52 организациях страны. Совались разного рода прикладные систем в авиации, космосе, экономике, АСУ, АСУ ТП

Метод сборки кибернетических, информационных и интеллектуальных ресурсов реализован Лаврищевой Е.М. в проекте РФФИ 16-01-00352 (2016-2018), был создан методом сборки экспериментальный вариант ядра OS Linux с участием студентов МФТИ, систем Grid Ethics и стандарта IEEE 828-2012 в итоговом отчете НИОКР 11020510073 от 5.02.16 N 16-01-00352 согласно теории и методов моделирования изменяемых (вариабельных) программных и технических систем». Рубрика 20.01.07. 002.6-027.21; 002.6.001.8 от 05.03.19.

. Обобщением разных вариантов операций сборки модулей стали фабрики программ в индустрии прикладных систем за рубежом и в СССР [13, 14]:

- Product Line/Product Family – конвейерная сборки ресурсов (К. Pochl).
- Мульти сборка генерации программ и данных (К. Czarnecki, A.Ezenaker).
- IBMSphere AppFab (<https://www.ibm.com/developerworks/ru/websphere/newto/>).
- Поточковая сборка разноязычных модулей Дж. Гринфильд and Г.Ленц.
- Сборка разноязычных модулей в ЯП (C, C++, Basic, Fortran и др.) И.Бей.
- Непрерывная интеграция (сборка) модулей (Continuous integration) М. Фаулер.

- Фабрика ПО Веб-служб (p&h) GAX, сборка SAP NetWeaver и др.
- Конвейерная сборка технических и программных средств В.М.Глушкова.
- Сборка КПИ, модулей <http://programsfactory.univ.kiev.ua>,
- Технология программирования со сборкой <http://7dragons.ru/ru> ИСП РАН и ГДР.

Разработаны зарубежные и отечественные технологические средства:

- 1) АСКОН (с 1989) включает следующие элементы:
- 2) - КОМПАС-3D для трехмерного моделирования на основе CAD / CAM / CAE;
  - ЛОЦМАН:PLM – система управления данными и ЖЦ;
  - а- ВЕРТИКАЛЬ – автоматизированное проектирование ТП.
- 3) ЗАО Топ Системы > (1992) - адаптированный вариант T-FLEX документооборот;
  - САМ – линейка ПП (Project Studio CS) в AutoCAD для станков с ЧПУ;
  - Autodesk Inventor для чертежей продуктов в ЕСКД и др.

Под управлением Лаврищевой Е.М. разработаны в рамках проектов РФФИ 352 и 206 средства обеспечения надежности, качества, защиты данных и безопасности систем (2016-2021).

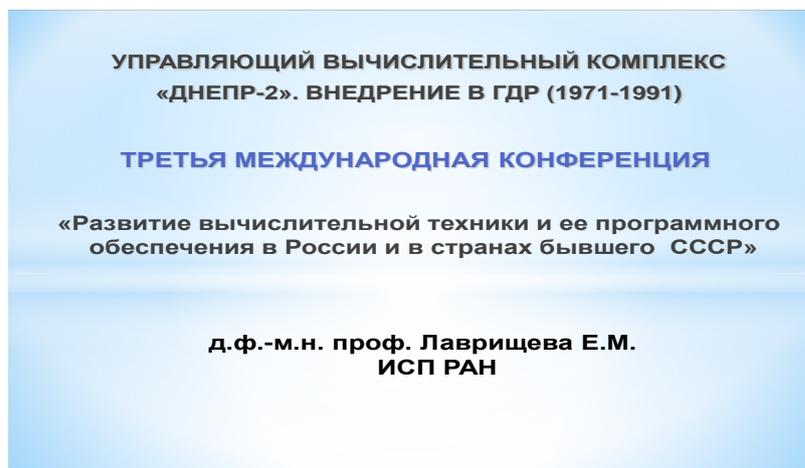
Метод сборки ресурсов для прикладных систем знаний (физика, математика, медицина, биология) представлен в доклад: <https://videonauka.ru/stati/30-metodika-prepodavaniya-tekhnicheskikh-distiplin/238>. и в докладе академика В.М.Глушкова.

**НАУЧНЫЕ ИДЕИ  
СОЗДАНИЯ ЭВМ, ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ,  
АСУ И ОГАС В СССР АКАДЕМИКА ГЛУШКОВА В.М.**



**Доклад  
д.ф.-м.н. проф. МФТИ, ГИС ИСП РАН  
Лаврищева Е.М.  
2023**

В презентации представлены создаваемые на ЭВМ системы программирования (трансляторы, компиляторы, отладчики, верификаторы, тестировщики, интеграторы, сборщики) и на УВК ДНЕПР-2 по договору с ВМНВ ГДР и внедрении СП АКД, АЛГАМС и КОБОЛ, и первую АСУ ТП для медицинских приборов под рук. В.М. Глушкова (см. Презентация 1975-1978).



Далее приводится базовый фундамент Вариантов 1, 2 метода сборки, представленные во введении.

### 1. Сборка модулей в системе АПРОП 1975-1990 (1 вариант)

Главное новшество в сборке модулей был интерфейс (межмодульный, межъязыковой и технологический) [19] и библиотека БМИ (64 интерфейсных функций) преобразования типов данных с одного ЯП к другому средствами системы АПРОП. Первое определение конвейерной сборки программ по типу конвейерной сборки на автомобильном заводе Форда дал В.М. Глушковым после выступления на IFIP-1975 он сделал доклад на Ученом Совете ИК АН УССР в 1975г. [2]. В программах связывающим звеном станет операция связи через интерфейс, подобно связывающих болтов и гаек в конвейерной сборке автомобилей на фабрике Форда. Идея интерфейса при связи модулей намного опередила зарубежные разработки инженерии ПО и технологии. Язык типа MIP (Module Interface Language) мы определили в 1980 г. С помощью сборки и были собраны системы Прометей, Протва, Руза, Яуза и специализированные приборы для авиации, космоса и Морфлота. Главной доминантой связи и взаимодействия компонентов и объектов в современных глобальных и сетевых средах стал программный интерфейс API (Application programs Interface), интерфейс IDL (Interface Definition Language), научный интерфейс SIDL (Scientifically IDL), SDL, SDL для сборки научных артефактов, сервисов, КПИ, информационных, кибернетических и интеллектуальных ресурсов.

**Межмодульный интерфейс** - это описание данных, передаваемых и принимающих модулей, представленное в интерфейсном посреднике, выполняющим функции передачи, приема и преобразования нерелевантных данных между двумя модулями. Язык определения интерфейсов модулей MIP в системе АПРОП способствовал описанию в посреднике: типов данных (ТД) задаваемых операторами CALL ЯП входных и выходных параметров, преобразованию неэквивалентных ТД. На основе описания посредников генерировался исходный код интерфейса с использованием библиотеки БМИ для сборки ресурсов с помощью примитивных функций преобразования БМИ в классе ЯП ОС ЕС. Интерфейс сыграл важную роль при сборке разноязычных модулей в программные системы (ПС) и их семейства - СПС.

**Межъязыковой интерфейс** - совокупность средств и методов представления и преобразования структур и ТД ЯП с помощью алгебраических систем (с алгеброй операций и функций интерфейса) обеспечение взаимно однозначного преобразования отличий ТД в ЯП объединенных интерфейсным языковым посредником (например, преобразование матрицы по строкам в Фортране в матрицу по столбцам в PL/1 и обратно; символьных данных в целое и др.). Библиотека интерфейса БМИ из 64 примитивных функций была передана по актам внедрения в 52 организации СССР и широко использовалась разработчиками программ в разных ЯП [2, 19-21].

**Технологический интерфейс** - это совокупность методов и средств для взаимосвязи процессов и операций ТЛ жизненного цикла (ЖЦ) реализации сложных программ, включая нормативные, методические документы и формы (каркас ТЛ, формат документа ТЛ, язык связи процессов и др.). Эти документы применяются при контроле результатов процессов, оценки требований и обеспечения

показателей качества, внесения изменений в продукт на процессах ЖЦ и передачи его промежуточного состояния следующему процессу и др. Методика создания ТЛ предложена в 1984г. [22], апробирована и усовершенствована на шести линиях АИС «Юпитер-470» [23], разрабатываемой по договору с ВМФ СССР для автоматизации задач управления флота, флотилии и подводных лодок. Статья стала первой работой по формализации и применению ТЛ в проектах разработки крупных информационных систем. Дальнейшим развитием ТЛ - продуктовые линии (Product Lines) SEI ([http://sei.cmu.edu/productlines/frame\\_report/](http://sei.cmu.edu/productlines/frame_report/)) 2004 г.

Созданная концепция интерфейсов модулей была автоматизирована генератором интерфейсных модулей-посредников интерфейса в системе АПРОП (1975-1982г.) и книги

Е.М.Лаврищева, В.Н.Грищенко. Связь разноязычных модулей в ОС ЕС» (М.: 1982, Финансы и статистика, 127с.). Там же дано описание языка интерфейса и набора функции БМИ, которые использовались при описании межмодульных посредников. Эта часть системы АПРОП способствовала сокращению объема работ при сборке разноязычных программных элементов на основе графовой структуры систем и средств генерации интерфейсных посредников.

*Интерфейсу* была посвящена международная конференция «Интерфейс СЭВ-1987» [24]. На ней были представлены все виды интерфейсов в системе АПРОП. Организатор этой конференции ГКНТ СССР наградил коллектив разработчиков (Коваль Г.И., Коротун Т.М., Лаврищева К.М., Грищенко В.Н.) почетными грамотами. Намного позже в 1998-2001 годах появились зарубежные языки описания интерфейсов: API, IDL, SIDL и др. Они используются и теперь при создании новых ПС и семейств СПС из готовых программ, компонентов повторного использования (КПИ) и сервисов средствами новых языков API, SDL, WSDL, PDFL и др.

Роль технологии сборочного программирования определил академик А. П. Ершов после выступления Е.М. Лаврищевой в 1978 году в Новосибирске на конференции «Основы системного программирования» с докладом «Подход к промышленной технологии изготовления больших программ» на Всесоюзной Конференции по Программированию «Методы и средства системного программирования». В докладе А.П. Ершова «Научные основы доказательного программирования» на президиуме АН СССР (1984). отметил, что операции сборки базируются на интерфейсном посреднике, обеспечивающим преобразование новых данных, включая Большие данные - Big Data, информационные, интеллектуальные и сервисные ресурсы средствами стандарта ISO/IEC 11404 GDT, GPD к формату указанных сред Интернет. Появился стандарт конфигурационной сборки IEEE 828 Configuration для систем Интернет E-Sciences, Semantic Web, Grid, Etics. Создаваемые системы проверяются на качество, безопасность и защиту данных. Аспектам создания теории и технологии программирования посвящен статья-доклад автора в Дома ученых РАН в 1917г. Никольского Н.И. Приведем ее ниже.



В данной презентации отображены метод модульного и компонентного программирования сборочного типа и методы верификации, тестирования и обеспечения качества ПС и СПС на период 2017 года. Отдельные аспекты этого доклада опубликованы в статьях [1-20]. См. презентацию лекции в МФТИ и на конференции в 2018.

## СБОРОЧНЫЙ ПОДХОД. ОБЪЕКТНО-КОМПОНЕННЫЙ МЕТОД МОДЕЛИРОВАНИЯ СИСТЕМ

Четвертая научно-практическая конференция

“Актуальные проблемы системной  
и программной инженерии”

Москва- 2015

доктор физ.- мат. наук, проф. Лаврищева Е.М.

**Метод сборки модулей основывается на** операциях link, Call, примитивных функциях интерфейса и библиотеке БМИ для преобразования ТД ЯП (Алгол, Фортран, Кобол, Модула, Пролог и др.). В книге «АПРОП. -Глушков В..М., Стогний А.А., Лаврищева и др. Москва, 1976 и монографии «Связь разно языковых модулей в ОС ЕС, Москва, 1982. -127с., как общее математическое ПО, включенное в общесистемные среды - Sun Microsystems, IBM, Intel, General Electric, CDC, Oberon, Unix, .Net. Модули специфицировались я ЯП и содержали раздел описания паспорта, в котором описывались:

- название предметной функции или метода представления модульного ресурса;
- ID – идентификатор ресурса, КПИ;
- описание параметров (входных и выходных) в операторе link, make, assembly, config в MPL;
- операции вызова CALL/RPC/RMI/WSDL;
- необязательные атрибуты (дата, состояние, версия, право доступа, автор, дата создания),

Операции обработки передаваемых данных через интерфейсный посредник представлены в презентации ниже и в ряде статей с соисполнителем проекта РФФИ №352 Рыжова А.В.



### Теория фундаментальных и общих типов данных для применения в Big Data

Лаврищева Е.М., Рыжов А.Г.

Доклад на  
DAMDID/RCDL'2016  
October 11 – 14  
Доклад в МФТИ 09.09.2019

Лаврищевой Е.М.  
д.ф.- м. н. профессор,  
гнс ИСП РАН.

**Важным элементом сборки модулей является - интерфейсный посредник. В его основе** лежит:

- стандартизация объектов, их свойств и характеристик;
- разработка формализованных правил (механизмов) сопряжения стандартизованных объектов;

– средства автоматизации механизмов связи объектов через интерфейсы: межязыковой, межмодульный, межпрограммный, пользовательский (см. выше). Назначение интерфейса состоит в устранении отличий в:

- 1) языковых средствах описания ЯП;
- 2) описании каждого отдельного модуля;
- 3) способах реализации модулей системами программирования с ЯП.

К отличиям языковых средств ЯП относятся неодинаковость синтаксического и семантического описания типов данных в разных ЯП функциональных возможностей модулей. К ним относятся:

- механизм конструирования новых типов данных отсутствует в языках Фортран, ПЛ/1, Кобол и имеется в языках Паскаль, Ада, Симула-67, Модула-2, Си, Альфард, CLU [13, 37, 38, 69, 92];

- некоторые предопределенные типы отсутствуют в определенных ЯП (например, символьный тип в ЯП Фортран);

- представление некоторых предопределенных типов отличается в разных ЯП (логический тип в языке ПЛ/1 представлен как битовая строка);

- динамические типы данных отсутствуют в Фортране и Коболе и имеются в языках Паскаль, ПЛ/1, Ада, Симула-67 и др.;

- организация внешних файлов различна (ПЛ/1 допускает последовательную, индексно-последовательную и прямую организацию файлов, Фортран не обеспечивает индексно-последовательной организации файлов);

- дескрипторы для представления структурных типов данных имеются в некоторых ЯП и не требуются в Фортране и Коболе;

- представление некоторых структурных типов отличается в различных ЯП (массивы в Фортране располагаются по столбцам, в других ЯП – по строкам).

На конференции в Новосибирске (1978) «Перспективы развития в системном и теоретическом программировании» автор выступила с докладом «Подход к промышленной технологии изготовления больших программ из модулей». Описан аппарат формирования программных структур (агрегат, комплекс, пакет программ) с помощью математических операций (U, /, +, -).

Построена матрицы достижимости для доказательства смежности вершин в графовой структуре.

Этот метод излагался на ряде конференций и излагался журнале «Программирование» (1980, 1982) и в книгах<sup>6-9</sup>. Метод сборки модулей в отечественном программировании. Аналогично это проводится в работе Липаева В.В. «Технология сборочного программирования».-1992.М:-280с. и Евстигнеева «Теория графов в программировании».- 1996.- и др.

А.П.Ершов\* принял метод сборки и определил теорию ТП на Всесоюзной конференции «Технология программирования» (1984)<sup>12</sup>, отметив в качестве перспективных методов программирования - синтез, сборку и конкретизацию. «Синтезирующее программирование базируется на методе доказательного рассуждения о правильности программы. Сборочное программирование осуществляет построение программы из уже существующих (проверенных на правильность) готовых фрагментов программ (reuses) и сборку их в сложную структуру. Конкретизирующее программирование обеспечивает построение системы по универсальной модели для некоторой предметной области»....

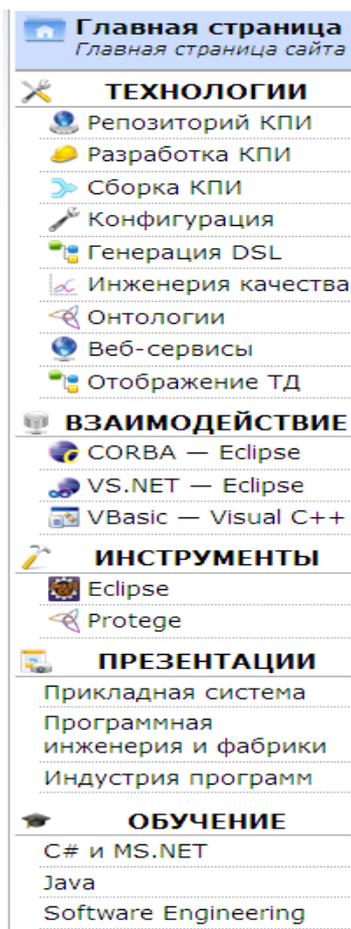
\*Ершов А.П. Введение в теоретическое программирование. М.: Наука, 1977. -288с.

Ершов А.П. Опыт интегрального подхода к актуальной проблеме ПО. - Кибернетика. -1984.

Ершов А.П. Научные основы доказательного программирования. – Доклад на звание академика в АН СССР, 1985. – с.1–14.

### **Инструментально-технологический комплекс (ИТК) для сборки ПС и СПС**

ИТК – задан сайтом <http://7dragons.ru/ru> («Программная инженерия. Технологии и обучение SE». В нем представлена технология разработки систем из готовых КПИ (сборка, конфигурация, онтология, качество, отображение данных и др.); взаимодействие между системными компонентами; операторы управления сайтом; онтология ЖЦ и вычислительной геометрии. Последний раздел ИТК - обучение «Программной инженерии», ЯП JAVA, C#, C++, Basic, Java. Разделы и подразделы даны на рис 2, а, в.



а)

Веб-сайт содержит разделы на рис. 2 а и перечень технологий.

#### ТЕХНОЛОГИИ ПОСТРОЕНИЯ ГОР и ПС:

Технология обслуживания репозитория КПИ,  
 Технология разработки КПИ,  
 Технология сборки КПИ,  
 Технология конфигурирования КПИ,  
 Технология генерации описания КПИ в языке DSL,  
 Технология оценка затрат и качества,  
 Технология онтологии вычислит. геометрии, ЖЦ ISO/IEC12207,  
 Технология веб-сервисов,  
 Технология генерация типов данных ISO/IEC 11404.

#### ВЗАИМОДЕЙСТВИЕ программ, систем и сред:

Модель CORBA - Eclipse-JAVA,  
 Модель VS.Net C# - Eclipse,  
 Модель Basic - C++.

#### ИНСТРУМЕНТЫ ИТК:

Система Eclipse,  
 Система Protégé.

#### ПРЕЗЕНТАЦИИ ПС В ИТК:

Система ведения зарубежных командировок для НАНУ,  
 Слайды про ИТК, фабрики программ,  
 Методологии построения ТЛ.

#### ОБУЧЕНИЕ:

Технологии программирования в C# VS.Net, JAVA,  
 Электронный учебник "Программная инженерия"  
 веб-сайт.

б)

Рис. 2. Главная страница веб-сайта ИТК в ИПС , ИСП РАН

#### К операциям сборки ПС и СПС из готовых ресурсов относятся:

- llnk ( $C_1 \dot{\cup} C_2 \dot{\cup} C_3$ );
- Reing ( $C_n$ ) – реинженерия ресурса;
- Revers ( $C$ ) – реверсная инженерия ресурса;
- Berification and validation (V&V) КПИ;
- client-server protocols Internet для обработки обмениваемых ресурсов;
- Creat classes ( $C_j$ ) – образование классов ресурсов.

#### К операциям обработки ресурсов относятся:

- depl - инсталляции (развертывания); del – удаление ресурса.
- addim – добавления реализации ресурса;
- addIn – добавления интерфейса ресурса; replIm – замещение реализации ресурса,
- replIm – замещение интерфейса и др.
- операции накопления, выбора модулей и ресурсов в Библиотеках и Репозиториях.

Данный сайт работал несколько лет Германий и настоящее время он в ИСП РАН и используется автором при чтении лекций в МФТИ.

Таким образом, сборочное программирование, используется в нашей стране представлено Лаврищевой Е.М. на сайте Глушкова в 1977г. (см. ниже).

[http://www.iprnet.kiev.ua/ef/nau\\_op\\_comp.htm](http://www.iprnet.kiev.ua/ef/nau_op_comp.htm)



*В.М. Глушков*

Теория и практика программирования

Научная деятельность

**От сборочного к компонентному программированию**

Идея Глушкова - фарбки программ  
Исследование ПИК и языков их описания  
Методы и принципы сборки программ с помощью интерфейса  
Генератор проблемно-ориентированных языков  
СОД  
Инициативы В.М. Глушкова на государственном уровне  
Технологическая подготовка разработки (ТПР)  
Становление программной инженерии  
Переход к компонентному программированию



Е.М. Лаврицева,  
зав. отделом  
программирования  
СКБ ММС ИК АН УССР,  
1977г.

## 2. Второй вариант сборки информационных, технических, операционных и интеллектуальных систем из ресурсов КПИ, Reuses, SOA, SCA Интернет

Второй вариант метода сборки создан с учетом новых средств Интернет – Семантик Веб, Grid и Etics создания информационных и интеллектуальных ресурсов Интернет с использованием семантических и сервисных ресурсов Интернет.

В 1990-х годах после развала СССР появились новые языки описания программ (IDL, API, ABI, WSDL, Python, Ruby, Snoop, Basic, C, C++ и др.) для описания модулей, компонентов, информационных, интеллектуальных, технических ресурсов появились новые операции сборки ресурсов в разных системных средах E-Sciences, Semantic Web Интернет:

**Link** IBM, MS.Net;

**Make** в BSD, GNU, MS Build;

**assemble, Kconfig, cmake** ресурсов в Grid, Etics;

**Cmake, assemble** SOA, SCA Semantic Web, Java, VS.Net;

**Weaver** BEA WebLogic Oracle, SAP Net\$

**building, confit** Bib-OGSWA, OML, JSON ASP и др. [7–12 ].

Приводится технология моделирования предметных областей знания средствами Grid.

### СЕМАНТИЧЕСКИЕ И СЕРВИСНЫЕ РЕСУРСЫ ДЛЯ РАЗРАБОТКИ ОНТОЛОГИИ НАУЧНОЙ И ИНЖЕНЕРНОЙ ПРЕДМЕТНЫХ ОБЛАСТЕЙ И ВЕБ- СИСТЕМ

Конференции

**Sciences and Information-2015 London**

**"Научный сервис в сети Интернет"**

19-24 сентября 2016, Абрау

**"Научный сервис в сети Интернет"**

19-24 сентября 2018, Абрау

Докладчик

Доктор физ.- мат.наук, профессор МФТИ  
Лаврицева Е.М.

В презентации показана система Grid для создания новых систем. Большую роль выполняла и система ETICS для реализации задач искусственного интеллекта (см. особенности презентацию).



ETICS\_Kov -Lavr  
2012 E-Science 2022-

## 2.1. Теория моделирование и программирование сложных систем научного представления знаний

После 1992г. появились новые информационные структуры данных (контейнеры, указатели, множества, списки, неструктурированные данные и др.) и стандарт GDT (ISO/IEC 11404 GDT -1998). Этот стандарт прошел многолетнюю апробацию и вышел новый вариант этого стандарта в 2012 г. В его состав вошли сложные типы данных агрегатного, параметрического, неструктурного, генеративного и сервисного типа, которые требуются для разработки новых интерфейсных примитивов в классе современных ЯП (C, C++, Basic, Python, Ruby, Java, WSDL и др.) для новых компьютерных платформ компьютеров и суперкомпьютеров.

Автор книги «Связь разно языковых модулей» провели доработку Библиотеки БИМ в направлении преобразования новых типов данных, включая Big Data и стандарты ISO/IEC 11404 GDT<sup>^</sup>-2007 и IEEE 828 Configuration-2003. В 2020 году ректор Сериков А.П. Интуит.ру привез Лаврищевой Е.М. в ИСП отформатированную книгу «Связь разноязыковую модулей ОС ЕС, 1982», которая использовалась в учебнике с 2007года «Методы и средства инженерии ПО» и в результате обучилось 998 студентов к 2016г. Этой книге требовалась 2 редакция с обработки Больших Данных – Big Data, поступающих с исследования недр земли, космоса и морей и Северного Ледовитого океана. 2022 году 30.08 вышла 2 редакция этой книги под названием: «Сборка модулей, объектов, и компонентов в ЯП.

2.Интерфейс фундаментальных и Стандартных типов данных для обмена данными между модулями.

5. Описание общих типов данных GDT, GLD. методом сборки ресурсов

5.1.Описание общих типов данных и конфигурационная сборка.

5.2.Метод сборки интеллектуальных и информационных ресурсов Интернет

5.3. Технология сборочного программирования общего назначения.

5.4. Общий сборщик ресурсов в Интернет и заявка на патент № 2020419438 от 15.08.2021 ФИПС..

6. Онтология - - Инструмент представления данных и знаний

6.1. Описание онтологии ЖЦ стандарта ISO/IEC 11404 в языке DSL и Proteger.

7.. Средства автоматизации программ в мировой практике.

7.1. Вариант экспериментального ядра OS Linux для прикладных систем областей знаний.

7.2. Пример сборки сервисных ресурсов для медицины.

Кроме того, добавлены средства обработки новых Big Data, обеспечения безопасности, защиты данных и качества и раздел Сборка экспериментального варианта ядра OCLinux с участием студентов МФТИ, включая работу с Data Mining, Reusebility Data (см. Ниже Презентацию с использованием Etics и Grid),



Документ  
Acrobat-2023.pdf

Далее рассматривается презентация сборочного создания экспериментального варианта ядра ОС Linux\* студентами МФТИ, которые написали магистерские работы и защитили их (Р. Газазулина, И.Ильин С., И.Губайдулин, Иванов А. и др.) (см. в списке литературы) .

Сборка вариантов ядра OS Linux приведена в презентации на конференции OS Day-2019. Лаврищева Е.М., Петренко А.К. Технология сборки интеллектуальных и информационных ресурсов Интернет. Научный сервис в сети Интернет: труды XXI Всероссийской

- научной конференции (23-28 сентября 2019 г., г. Новороссийск). — М.: ИПМ им. М.В. Келдыша, 2019. — С. 469-488. <https://doi.org/10.20948/abrau-2019-93>
46. Лаврищева Е.М., Зеленов С.В., Пакулин Н.В. Методы оценки надежности программных и технических систем. Труды Института системного программирования РАН. 2019; 31(5):95-108. [https://doi.org/10.15514/ISPRAS-2019-31\(5\)-7](https://doi.org/10.15514/ISPRAS-2019-31(5)-7)
47. Лаврищева Е.М., Мутилин В.С., Козин С.В., Рыжов А.Г. Моделирование прикладных и информационных систем из готовых сервисных ресурсов Интернет. Труды Института системного программирования РАН. 2019; 31(1):7-24. [https://doi.org/10.15514/ISPRAS-2019-31\(1\)-1](https://doi.org/10.15514/ISPRAS-2019-31(1)-1)



**Технология сборочного создания  
прикладных, технических и ОС  
для современных областей знаний  
в Интернет**

**Лекция  
Студентам МФТИ кафедры Информатики и  
прикладной математики 17 октября 2021года**

**Лаврищевой Е.М.  
д.ф.- м. н., почетного профессора МФТИ,  
г.н.с. ИСП РАН им. Иванникова В.П.**

Разработана и напечатана книга по моделированию и программированию современных предметных областей знаний (включая предыдущие сущности лекций презентаций на эту тему).



Монография Е.М.Лаврищева Е.М., Петров И.Б., Петренко А.К. Парадигмы моделирования и программирования задач и данных предметных областей знаний. - Москва. -ДиректМедиа, 2021. -495с. (см. приведенную 'не книге, 2021) в издательстве Юрайт on line [www.ura.it.ru](http://www.ura.it.ru), отображены многие аспекты отечественного и зарубежного пути развития и создания сложных прикладных систем областей знаний (физика, математика, медицина, биология, химия).

**2.2. Реализация технических систем методом сборки операционных, интеллектуальных и информационных систем (2 вариант).**

Используются средства Семантик Веб WWW3С и E-Sciences (Grid, ETICS) и Нано для создания интеллектуальных, информационных и технических систем для медицины, биологии. Они реализуют процессы управления научными знаниями, экономикой и практической реализацией работ коллективами путем;

- **отображения знаний** в Базах Знаний, предоставление их пользователям и разработчикам систем;

- **моделирования (knowledge modelling)** знаний для их использования при решении конкретных прикладных задач;

- **поиска и выбора** (knowledge retrieval) знаний из различных хранилищ в подмножество контента для релевантного решения конкретной прикладной задачи;

- **повторного использование** (knowledge reuse) знаний в виде готовых описаний, образцов (patterns), моделей в разнообразных контекстах;

- **публикации** (knowledge publishing) знаний в стандартизированной форме для последующего распространения;

- **обновления** (knowledge maintenance) и сохранения знаний в Базах Знаний;

- **приобретения** (knowledge acquisition) знаний для анализа неструктурированной **информационных средств** (тексты, изображения, сценарии и др.), преобразования неявных знаний в явные, решения задач обработки данных огромных объемов и сохранения их в БД.

Основу интеллектуализации информационных ресурсов (документов, текстов, таблиц, страниц и др.) составляют процессы:

- форматирование данных больших объемов в БД глобального масштаба с обеспечением сохранности, защищенности и безопасности;
- поиска, выбора фрагментов данных и их транспортирование по запросам пользователей;
- анализа, интеграции, агрегации информации (например, документов, в требуемые структуры для обработки на разных уровнях управления организациями страны и др.) и проведения вычислений задач в физике, математике, медицине и др. (см. презентацию)

## **ТЕОРИЯ МОДЕЛИРОВАНИЯ ИНФОРМАЦИОННЫХ И ИНТЕЛЛЕКТУАЛЬНЫХ ЗАДАЧ ПРЕДМЕТНЫХ ОБЛАСТЕЙ ЗНАНИЙ**

Лаврищева Е.М.

почетный профессор МФТИ, гнс ИСП РАН, lavr@ispras.ru

Международная научная конференция “Численное моделирование в  
механике сплошных сред” памяти О.М. Белоцерковского 27- 29  
октября 2021

Доклад

Е.М.Лаврищевой

д.ф.-м.н., профессора, гнс. ИСП РАН

**Интеллектуализация знаний средствами Grid, Ethics. Semantic Web**

**Система GRID** ([www.globous.org](http://www.globous.org)) обеспечивает взаимодействие ресурсов через операции RPC/RMI assembling, confit, make и устанавливают связь ресурсов между собой при работе Cloud Computing и Big Data. Система GRID (рис. 8) управляет программными, сервисными и др. ресурсами. Сборка разнородных сетевых ресурсов в системы, приложения пакеты, работающие с данными разного объема проводится в системе GRID и Etics (рис.9). Ресурсы описываются в разных ЯП. Базовые сущности системы, пакета, их связи и результаты описываются согласно CIM (Common Information Model). В системе Etics стандартизовано описание ТД для главных объектов: *Проект, Подсистема и Компонент*. *Проект*. Подсистема может содержать только Компоненты. В системе предложена модель данных CIM для связей между разными объектами.

*Модель данных*, как и модель CIM, позволяет вводить формальные сущности в структуре проектов, описывать объекты и связи между ними, а также предоставлять результаты выполнения. Внутреннее сохранение данных базируется на модели данных реляционного типа, реализованной в MySQL

В процессе создания *моделей знаний* предметных областей используются система KBS (Knowledge-based systems), CommonKADS и др. Они обеспечивают построение библиотек знаний, содержащих элементы решения задач, которые затем могут повторно использоваться и в других областях знаний.

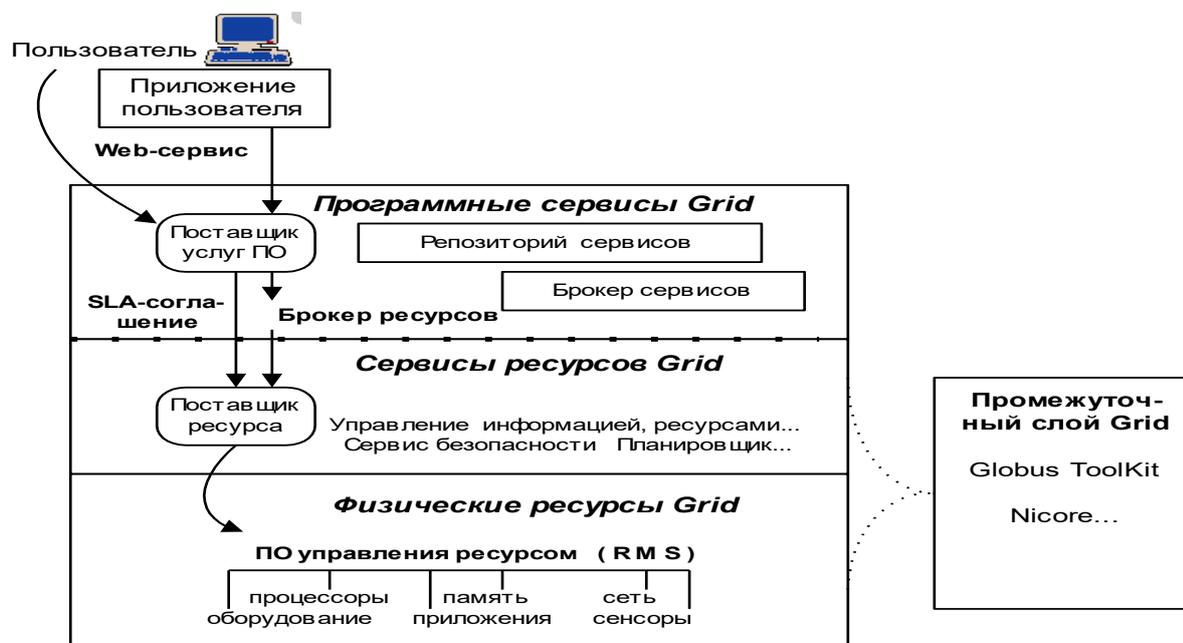


Рис.2. Общая структура функционирования GRID

К современным средствам представления знаний в терминах дескриптивной логики относятся: *FaCT* (на LISP), *FaCT++* (на C++), *CEL*, *KAON-2* (JAVA), *MSPASS* (C) и *Pellet* (JAVA) и др. Необходимой составляющей представления знаний являются *прикладные web-сервисы*, ориентированные на коллективное решение задач в предметных областях знаний в современных средах (problem-solving environments, PSE) в мировом сообществе.

Исходя из того, что программа на ЯП – это некоторый формальный или математический текст, то предложена система управления текстовой информацией и *терминологией* (System Quirk – SQ, <http://www.computing.surrey.ac.uk/SystemSQ>). Эта система ориентирована на создание и ведение терминов в терминологической базе данных (ТБД) и баз знаний (БЗ), а

также организацию коллекций текстов на компьютере с помощью инструментов Virtual Corpus, KonText, Ferret, Grid и др.

Для ведения больших объемов данных и обеспечения функционирования систем в глобальной сети Интернет используются системные, сервисные и коммуникационные ресурсы, включая средства Семантик Веб.

*Семантик-Веб* — эта информационная среда в World Wide Web, которая предоставляет семантические ресурсы, языки, средства и инструменты разработки онтологий предметных областей знаний, прикладных систем и бизнес-процессов с использованием накопленных знаний в среде <http://semwebprogramming.org>.

В этой среде проводится автоматизированная обработка научных задач, больших данных в разных форматах, интеграция данных из коллажей (Mash-Ups), поиск и конфигурирование веб-сервисов, управление интеллектуальными агентами в мобильных приложениях и др.

Одним из инструментов интеллектуализации знаний в Семантик Веб о разных предметных знаниях (математики, физики, биологии, медицины и др.) является онтология ([www.semantic.web.ontology](http://www.semantic.web.ontology)).

Онтология представления знаний об области знаний лежит понятийная база, совокупность концептов (понятий) и отношений между ними, классификация понятий и их таксономия в виде тезаурусов, а также методы создания профильных онтологий. Онтология является инструментом построения концептуальной модели для областей знаний/доменов, которая включает объекты, классы объектов, структуры данных, связи и отношения (теоремы, ограничения) применительно к конкретной области знаний. Имеется опыт разработки онтологии домена SE - ЖЦ ISO/IEC 12207 -2007 в языке OWL(Web Ontology Language) в среде Protégé 2.3 с выходным результатом в XML. Подход к автоматизации ЖЦ докладывался на Международной конференции «Science and Information-2015» ([www.conference.thesai.org](http://www.conference.thesai.org)) в Лондоне и на XVII Всероссийской научной конференции-2015, 2016 «Научный сервис в сети Интернет» в г. Новороссийск, ИПМ им. М.В. Келдыша [18-20, 29, 33].

Для применения больших объемов информации (Big Data) при решении научных задач предметной области могут использоваться словари и концептуальные модели и приложений предметных областей знаний с помощью предметно-ориентированных языков (OWL, DSL и др.), инструментов FODA, Protégé, DSL Tool, KBuild, Kconfig (<http://www.Sap.org>) и др. См. презентацию доклада в Лондоне на конференции Sciences and Information-2015 [30-35].



**Ontological approach to the formal specification of the Standard Life Cycle ISO/IEC 12207**

Science and Information Conference-2015,  
26-31 July, London



**E.M. Lavrischeva,**  
doctor of phy. and math. sciences,  
Honoured worker of science and  
technique Ukraine, prof. MIPT, Main  
scientist of ISP RAS

## CASE инструменты сборки программ с Big Data

В настоящее время используются средства поддержки сборки (интеграции) разных программ: **Make, Apache Ant, Apache Maven, Gradle** и др.

**Make** — это кросс-платформенная система автоматизации сборки ПС из исходного кода. Она генерирует файлы управления сборкою, например, Makefile в системах Unix для сборки посредством операции make.

**Apache Ant** — JAVA-утилита для автоматизации процесса сборки программного продукта. Ant — платформо-независимый аналог UNIX-утилиты Make, но с использованием языка JAVA приспособленный для JAVA-проектов. Самая важная непосредственная разница между Ant и Make состоит в том, что Ant использует XML для описания процесса сборки и его зависимостей, тогда как Make имеет свой собственный формат Makefile. По умолчанию XML-файл, называемый build.xml, осуществляет сборку.

**Apache Maven** — программный инструмент для управления (*management*) JAVA-проектами методом сборки (*building*) ПО аналогично Apache Ant, но с более простой build-моделью конфигурации, которая базируется на формате XML. Двигатель ядра может динамически загружать плагины с репозитория, который обеспечивает доступ ко многим версиям разных JAVA-проектов с открытым кодом Apache.

**Gradle** — система автоматической сборки, построенная на принципах Apache Ant и Apache Maven, но в отличие от них представляет DSL на языке Groovy вместо традиционной XML-подобной формы представления конфигурации проекта.

### 2.3. Задачи преобразования типов данных для общего сборщика в Интернет

При преобразовании сложных неструктурированных данных решаются задачи:

- специфицирование внешних ТД объектов в WSDL, сохранение их в БД и в репозиториях;
- преобразование ТД в новых ЯП<sub>1</sub>, ..., ЯП<sub>n</sub>;
- реализацию ТД GDT к виду специальных примитивных функций FDT;
- представление ТД FDT к виду специальных примитивных функций и формату Фреймворка;
- эквивалентные отображения и генерацию данных GDT $\Leftrightarrow$ FDT с учетом платформ современных компьютерных и кластерных систем Интернет где ресурсы будут работать..

Полуструктурированные, неструктурированные и расширяемые ТД используются в информационной среде, в связи с исследованиями недр земли, космоса и океана. Потребуется практическая обработка этих новых структур данных, чтобы с ними решались эффективно прикладные задачи в разных прикладных областях, особенно работающих с Big Data.

### Подходы к обработке неструктурированных данных Big Data

**Big Data** — набор данных большого объема, а также средств, инструментов и методов представления неструктурированных огромных объемов данных для получения данных и эффективного использования их на многочисленных узлах сети Интернет при решении прикладных Intelligence ИС и ИТ систем с использованием СУБД [16].

Для работы с большими объемами данных сформировался метод ETL (Extract Transform Load), с помощью которого производится:

- извлечение данных из внешних источников;
- трансформация и очистка данных с учетом требований;
- загрузка данных в хранилища данных;
- анализ данных и их перенос из одного приложения в другое и др..

К основным свойствам Big Data относятся:

– *горизонтальная масштабируемость* обрабатываемых больших данных и большого количества кластеров и серверов;

– *отказоустойчивость* по отношению к сбоям на **процессорах** кластеров и в узлах сети. Так, инструмент Hadoop-кластер Yahoo имеет более 42000 компьютеров, среди которых часть из них может выходить из строя;

– *локализация данных и обработка* их на серверах, где практически хранятся большие данные для решения соответствующих задач;

– *изменение числа работающих* на кластере с помощью средств MySQL Cluster. Большие данные могут быть представлены как tensors, которые управляют вычислением (например, полилинейное обучение подпространств Multilinear Subspace Learning и др.).

К системным средствам обработки Big Data относятся:

NoSQL-БД нереляционные и распределенные данные с открытым кодом и горизонтальной масштабируемостью, эффективно поддерживают случайное чтение, запись и версиюность.

MapReduce — модель распределённых вычислений, которая используется при параллельных вычислениях с большими данными в компьютерных кластерах.

Hadoop — свободно распространяемый набор утилит, библиотек и фреймворков для разработки и выполнения распределённых приложений (в том числе, MapReduce-программ), работающих на кластерах с сотнями и тысячами узлов.

СМБ — системы обработки больших данных в рамках Cloud-вычислений.

Big Data анализируются средствами: статистических и динамических методов анализа искусственного интеллекта, нейронных сетей, математической лингвистика; A/B Testing, Crowdsourcing Data Fusion; Integration Genetic Algorithms Machine Learning; Natural Language Processing; Signal Processing Simulation and Visualization; Massively Parallel Processing; Search-Based Applications, Data Mining, Multilinear Subspace Learning и др. (см. презентацию).

## ПУТИ РАЗВИТИЯ ИНДУСТРИИ ПРОГРАМНОЙ ПРОДУКЦИИ по Глушкову 2013



Доклад в КНУ  
доктора физ.-мат. наук, профессора  
**Е.М.Лаврищевой**

Лекция в МФТИ  
2021

Рис.4 Глушков и АСУ и АСУТП

Согласна приведенной презентации В.М. Глушков является основоположником сборки модулей и систем, начиная с 1972 годв и до того момента, когда заболел в больнице ЦК... Можно посмотреть, например, фабрику программ GDM (Generative Domain Model) и др.

#### Использование больших данных (Big Data, 2002) при решении задач

При исследовании недр земли, океанов и небесного пространства, космоса накапливаются много данных, которые запоминаются в альтернативных СУБД и используются для решения разнообразных задач E-science, Bisness Intelligence, экономики.. Для работы с большими данными имеется метод ETL (Extract Transform Load), включающий извлечение данных из внешних источников, их трансформацию и очистку для удовлетворения требованиям бизнес-моделей и загрузки таких данных в Хранилище.

#### Методы обработки Big Data

1. Горизонтальная масштабируемость состоит в обработке расширяемых больших данных, в связи с ростом объёма данных, количество кластеров и серверов в 2 раза.

2. Отказоустойчивость масштабируемости подразумевает, чтобы в кластере было много компьютеров и они должны учитывать возможные сбои и аварии. Hadoop-кластер Yahoo имеет более 45000 компьютеров и часть из них может выходить из строя.

3. Локализация данных для обработки данных на том же компьютере сервера, на котором хранятся Big Data.

4. Изменение числа работающих на кластере компьютеров средствами MySQL Cluster.

Объекты обработки в Big Data:

- искусственный интеллект, искусственные нейронные сети;
- предиктивная аналитика, системный анализ;
- статистические методы, краудсорсинг;
- математическая лингвистика, Теория ТД и др.

**Cloud Computing и Big Data** - это парадигма распределенного широкомаштабного компьютера, в котором набор абстрактных виртуальных, динамически масштабированных управляющих машин, памяти, платформ и сервисов, задаваемых через Интернет:

- генетические хранилища (долго, 100% точность восстановления, в 1 губ – 2000 ТБ);
- IBM 154 ТВ (размер в 62 раза больше, чем LTO-6, технология Nano cubic) и др.

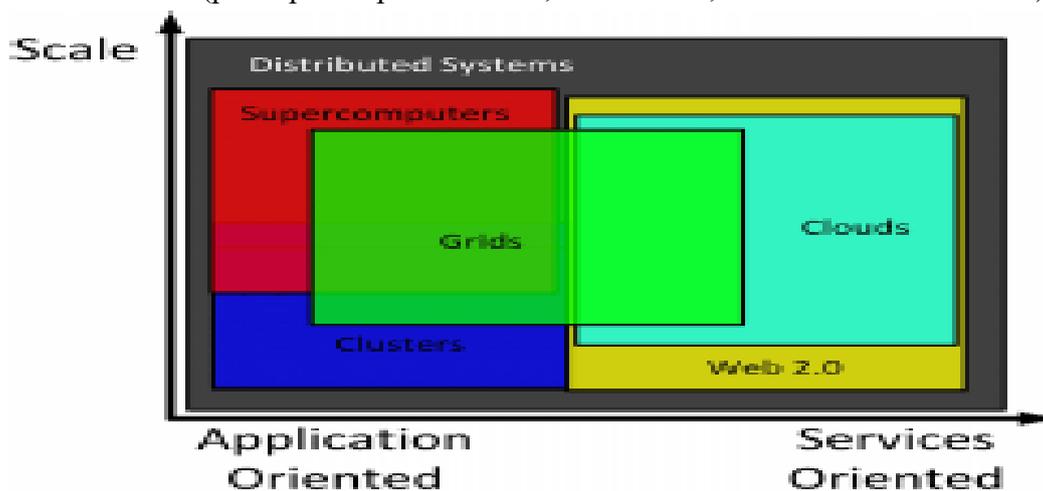


Рис.5. Cloud Computing

Cloud computing для информационных технологий с SaaS (Software as a Service) для удаленного применения сети VPN (Virtual Private Network) с многочисленными WEB-сервисами с высоким уровнем надежности, безопасности и технологии виртуализации IaaS, PaaS и SaaS.

Cloud содержит такие виды услуг: testing-as-a-service, security-as-a-service, database-as-a-service, process-as-a-service, application-as-a-service, platform-as-a-service и UC-as-a-service.

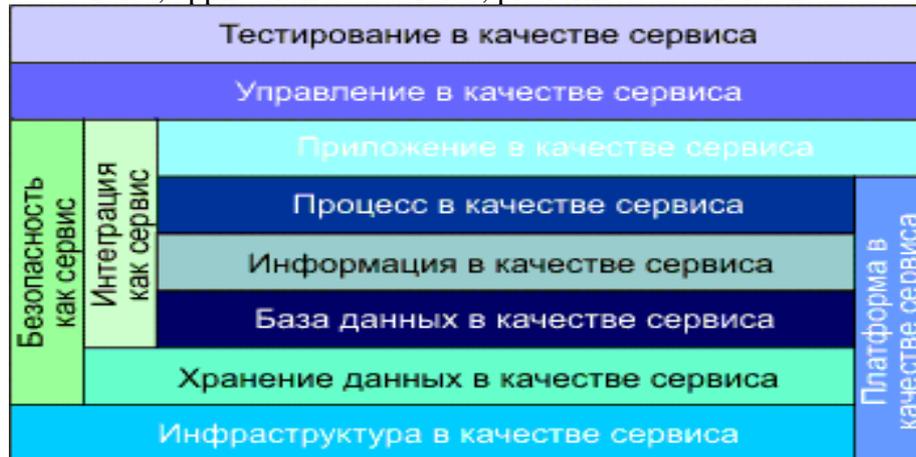


Рис.б. Тестирования элементов сервиса

**Cloud computing** содержит 15 видов услуг для проведения вычислений:

- сохранение данных (storage-as-a-service),
- базы данных (database-as-a-service),
- информация (information-as-a-service),
- процесс (process-as-a-service),
- приложения (application-as-a-service = software-as-a-service),
- платформа (platform-as-a-service),
- интеграции (integration-as-a-service),
- информационные технологии (IT-as-a-service).

Программист может использовать весь спектр услуг у одного провайдера и у других провайдеров.

#### **Сервисные услуги Cloud Computing Internet:**

ABI - Application Binary Interface включает:

- набор инструкций процессора: структуру регистрового файла, организация стека, способы доступа к памяти и т.д.;
- размеры, расположение, выравнивание базовых ТД, с которыми работает процессор компьютера;
- соглашение о вызовах (операции передачи аргументов и возвращаемых значений процедур) на стек, регистры процессора.

ABI операционной системы включает:

- правила задания системных вызовов кода приложения;
- бинарный формат объектных файлов, библиотек, исполняемых файлов.

### **2. 3. Идея общего сборщика ресурсов в Интернет**

В качестве общего вывода по рассмотрению операций сборки для разных сред можно сделать вывод о том, что приведенные операции сборки (link, make, confit, assembling, building, weaver, integration) ресурсов в системных средах, являются базовыми средствами для создания общего «сборщика» готовых ресурсов для прикладных, сервисных, информационных и технических систем в сети Интернет.

Приведенные операции сборки (link, make, confit, assembling, weaver) ресурсов в заданных системных средах имеют схожие операционные способы сборки разных вариантов ресурсов: модулей в разных ЯП, исходных и выходных текстов компиляторных программ, сервисных и системных ресурсов Веб среды Интернет, технических средств компьютеров имеют похожие

способы сборки, которые повторяются в разных общесистемных средах. Как бы не назывались способы сборки семантика сборки остается одинаковой. Способ сборки зависит от данных, которые используются при обмене данными между связываемыми разнородными ресурсами.

Рассмотренные средства генерации общих типов данных стандарта ISO/IEC 11404 GPD – 2012 требуют создания набора новых примитивных функций для нестандартных типов данных (портфелей, контейнеров, шаблонов, указателей, неструктурных данных и др.) и использоваться в названных способах сборки разнородных ресурсов Интернет.

Для создания общего «сборщика» готовых ресурсов в прикладные и технические (макро конвейерные) системы в Интернет, ставится задача создания сборщика в виде:

1. Определить формальный вариант задания операции сборки ресурсов
2. Создать библиотеку примитивных функций для всех данных систем Интернет и дорабатывать примитивные функции БМИ для новые ТД (контейнеров, шаблонов и др.) согласно стандарта GPD.
3. Создать анализатор (агент) операции сборщика и взаимодействия (компиляторных, системных, прикладных, технических) с учетом всех типов ресурсов.
4. Проводить анализ соответствия типа ресурса и взаимодействия с другим ресурсом через интерфейс и осуществлять обращение к соответствующим программам преобразования обмениваемых данных для генерации соответствующего преобразования по теории преобразования типов данных стандарта GPD.
5. Управлять сборкой при выполнении всех процессов, определенных в стандарте IEEE 828- Configuration:2006;
6. Выдавать сведения о результатах сборки при завершении работы сборщика.
7. Обеспечить размещение ресурсов в библиотеках и хранилищах Интернет из разных предметных областей знаний (медицины, биологии, генетики, математики и др.).

Техническим результатом общего сборщика является:

- 1) простота поиска ресурсов в хранилищах Интернет и снижение затрат на разработку за счет готовых правильных многообразных ресурсов и настройку их на конкретные условия применения;
- 2) повышение качества и производительности создаваемых из ресурсов Веб-приложений и систем за счет системных операций замены отдельных более правильных ресурсов и изменения конфигурации (архитектуры) варианта файла с получением качественных решений функциональных задач приложений в разные предметные области знаний.

**Сборщик интеллектуальных, информационных и кибернетических ресурсов в Интернет (Заявка в ФИПС в 17.12.2020)**

### **Формула изобретения**

1. Способ сборки программных и информационных ресурсов сборщиком Интернет, включающий определение формального варианта спецификации операции сборки ресурсов, развитие библиотеки примитивных функций для всех общесистемных сред Интернет и обращения к соответствующим функциям преобразования обмениваемых данных ресурсов и генерации необходимых значений, данных согласно общей теории типов данных стандарта ISO/IEC 11404 GDT, **отличающийся тем, что** дорабатывают примитивные функции библиотеки для новых стандартных и неструктурированных типов данных; создают анализатор элементов операции сборки связываемых компиляторных, системных, прикладных, технических и сервисных ресурсов; проводят анализ соответствия типа ресурса и взаимодействия с другим ресурсом через интерфейс; управляют конфигурационной сборкой, выполняя все ее процессы, определенные в стандарте IEEE 828 Configuration, и

выдают сведения о результатах сборки, тестировании, оценки качества конфигурации и завершении работы сборщика Интернет.

2. Способ в п. 1 отличается тем, что под новым типом данных имеют в виду стандартные элементы в виде контейнеров, шаблонов, портфеля, массивов, таблиц, интерфейсов готовых программных, технических, информационных типов данных стандарта ISO/IEC 11404 и нестандартных типов данных, основанных на XML (отчеты, сценарии, указатели, ссылки, двоичные BLOB с датчиков, агрегаты и др.).

3. Способ в п. 1, отличается тем, что созданный анализатор элементов операции сборки осуществляет синтаксическую и семантическую проверку правильности элементов и устанавливает связь между элементами операции сборки через интерфейс для выбора программы преобразования передаваемых типов данных другому ресурсу из Библиотеки с учетом типов данных стандарта ISO/IEC 11404 GDT.

4. Способ сборки по п.1 отличается тем, что проводится реализация стандартных и нестандартных типов данных (сценариев, отчетов, агрегатных данных с датчиков и др.) в виде примитивных функций для размещения их в библиотеке интерфейсных функций и их использования при последующей сборке и обработки конфигурационной архитектуры из готовых ресурсов.

#### 2.4. Обеспечения качества создаваемых из ресурсов сложных систем

Согласно стандарту ISO/IEC 12207 ЖЦ ПО регламентируется планирование, управление качеством и оценку затрат на создание системы. На этих процессах ЖЦ проводится анализ достижения качества; верификация и валидация (V&V) ресурсов и оценивание степени достижения отдельных показателей качества; тестирование готовой системы; сбор данных об отказах, дефектах и др. ошибках; оценивание надежности по соответствующим моделям надежности с учетом результатов тестирования [33, 35, 36].

**Модель качества стандарта ISO/IEC 9000 (1-4) Quality** задает шесть показателей (характеристик)  $q_1$  —  $q_6$  ( $q$  — quality) качества:

- $q_1$  — функциональность (functionality),
- $q_2$  — надежность (reliability),
- $q_3$  — удобство (usability),
- $q_4$  — эффективность (efficiency),
- $q_5$  — сопровождаемость (maintainability),
- $q_6$  — переносимость (portability).

Каждая характеристика  $q_i$  рассчитывается по специальным формулам и метрикам стандарта. Надежность оценивается согласно полученных на процессе тестирования ошибок, дефектов и отказов в ПО и по соответствующим моделям надежности (оценочным, измерительным и др.). Данные по всем показателям качества  $q_1$  —  $q_6$  оцениваются по формуле:

$$q_i = \sum_{j=1}^6 a_{ij} m_{ij} w_{ij}$$

где  $a_i$  — атрибуты каждого показателя качества ( $i = 1..6$ );  $m_i$  — метрики каждого атрибута качества;  $w_i$  — вес каждого атрибута показателя качества системы. Полученные значения по показателям модели качества входят в сертификат качества продукта. Варианты оценки показателей качества сконфигурированных систем приведены на сайте ИТК [18, 19].

# Модельный подход к обеспечению безопасности и надежности Web-сервисов

Е.М. Лаврищева  
С.В. Зеленов

ИСП РАН

Приведенный метод обеспечения безопасности и надежности используется на практике в авиации и космосе и флоте.

## 2.5. Заключение

Технология сборки Web-систем из интеллектуальных и сервисных ресурсов сформировалась в рамках проекта РФФИ №16-01-00352 «Теория и методы разработки изменяемых программных систем» [21–23]. В данной работе рассмотрены базовые понятия конфигурационной сборки систем, веб-систем из готовых ресурсов — КПИ, reuses, service-components, Web-services, которые описываются в современных ЯП (C, C++, JAVA, Python, Ruby, Basic и др.), а их интерфейсы в языке WSDL, APL. Приведены операции сборки систем из готовых сетевых ресурсов. Дан анализ подходов к определению интеллектуальных элементов в среде Семантик Веб и открытых моделей SOA и SCA для представления системных сервисов, серверов, клиентов и данных Big Data в Cloud Computing.

Отработан метод сборки модулей, объектов и компонентов с помощью операций (link, make, smake, config, weaver) в рамках действующих систем АПРОП, BSD, GNU, MSBuild; EuroGrid; Semantic Web; weaver BEA WebLogic Oracle, SAP Net и др.). Приведен теоретический аппарат сборки ресурсов с использованием фундаментальных и общих типов данных и функций преобразования передаваемых неэквивалентных, неструктурированных данных (включая Big Data) в среде Интернет на основе стандарта ISO/IEEE 11404 GDT. Описана конфигурационная сборка ресурсов и процессов верификации, тестирования и оценки качества с помощью стандарта ISO/IEC 9000 (1-4) Quality полученного ПП.

В рамках проекта РФФИ 16-01-00209 (2019–2021) реализована клиент-серверная архитектура систем средствами интеллектуализации готовых ресурсов (системных, сервисных, информационных и интеллектуальных) с обеспечением безопасности, защиты данных и качества ресурсов в Cloud Computing и Big Data.

## Литература

1. Лаврищева Е.М., Грищенко В.Н. Связь разноразличных модулей в ОС ЕС. — М.: Финансы и статистика. — 1982. — 136 с.
2. Лаврищева Е.М. Интерфейс в программировании // Проблемы программирования. — Киев: 2006. — №2. — С.126–139; Formal fundamentals of component interoperability in programming / Lavrischeva K.M. // Cybernetics and Systems Analysis. — 2010. — Volume 46. — Issue 4. — P. 639–652. — doi:10.1007/s10559-010-9240-z
3. Лаврищева Е.М., Грищенко В.Н. Сборочное программирование. — Киев: Наук. думка, 1991. — 236 с.
4. Липаев В.В., Позин Б.А., Штрик А.А. Технология сборочного программирования. — М.: Радио и связь, 1992.
5. Лаврищева Е.М., Грищенко В.Н. Сборочное программирование. Основы индустрии программных продуктов. — Киев: Наук. Думка, 2009. — 371 с.

6. Ершов А.П. Опыт интегрального подхода к актуальной проблеме ПО // Кибернетика. — 1984. — С. 11–21; Научные основы доказательного программирования // Вестник АН СССР. — 1984. — № 10. — С. 9–19.
7. Web Services Description Language (WSDL) 1.1/W3C Note 15 March 2001. — URL: <http://www.w3.org/TR/wsdl>.
8. Semantic Web. — URL: <http://www.w3.org/2001/sw/>.
9. Reference Model for Service Oriented Architecture 1.0. — URL: <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.html>
10. Web Services Resource 1.2 (WS-Resource). — URL: [http://docs.oasis-open.org/wsrif/wsrif-ws\\_resource-1.2-spec-os.pdf](http://docs.oasis-open.org/wsrif/wsrif-ws_resource-1.2-spec-os.pdf).
11. SOAP Version 1.2 Part 1: Messaging Framework (Second Edition) // W3C Recommendation. 27 April 2007. — URL: <http://www.w3.org/TR/SOAP12-part1>.
12. Semantic Web programming / S. Hebel, M. Fisher, R. Blace, A. Peter-Lopes. — Willey Publishing Inc., 2008. — URL: <http://semwebprogramming.org>.
3. Лаврищева Е.М. Теория и практика фабрик программ // Кибернетика и системный анализ. — No. 6. — 2011. — С. 145–158; Theory and practice of software factories / K.M. Lavrischeva // Cybernetics and Systems Analysis. — 2011. — Volume 47. — Issue 6. — P. 961–972.
14. Е.М. Lavrischeva. Classification of software engineering disciplines // Cybernetics and Systems Analysis. — 2008. — Volume 44. — Issue 6. — P. 791–796.
15. Лаврищева Е.М. Теория объектно-компонентного моделирования программных систем // Препринт ИСП РАН 29, 2016. — С. 1–52. — URL: ISBN978-5-9-474-25..
16. Иванников В.П., Дышлевый К.В., Мажелей С.Г и др. Распределенные объектно-ориентированные среды // Труды Института системного программирования РАН. — Том 1. — 2000. — С. 100–121.
17. Лаврищева Е.М. Рыжов А.Г. Применение теория общих типов данных стандарта ISO/IEC 12207 GDT применительно к Big Data // Conference “Actual problems in science and ways their development”, 27 December 2016, <http://euroasia-science.ru/>. — С. 99–110.
18. Е. М. Лаврищева, Л. Е. Карпов, А. Н. Томилин, Системная поддержки решения бизнес-задач в глобальной информационной сети, Научный сервис в сети Интернет: Труды XVII Всероссийской научной конференции (19–24 сентября 2016 г., г. Новороссийск). — М.: ИПМ им. М.В. Келдыша, 2015. — с.101–127.
19. Лаврищева Е. М., Карпов Л. Е., Томилин А. Н. Семантические ресурсы для разработки онтологии научной и инженерной предметных областей // Научный сервис в сети Интернет: Труды XVIII Всероссийской научной конференции (19–24 сентября 2016 г., г. Новороссийск). — М.: ИПМ им. М.В. Келдыша, 2016. — с.126–138. — doi:10.20948/abrau-2016-8.
20. Лаврищева Е. М., Карпов Л. Е., Томилин А. Н. Подходы к представлению научных знаний в Интернет науке // Сб. XIX Всероссийской научной конференции «Научный сервис в сети Интернет», Новороссийск, 18–23 сентября 2017. — С. 310–326.
21. Лаврищева Е.М. Компонентная теория и коллекция технологий для разработки промышленных приложений из готовых ресурсов // Труды 4-ой научно-практической конференции «Актуальные проблемы системной и программной инженерии», АПСИИ-2015, 20–21 мая 2015. — С. 101–119.
22. Лаврищева Е.М. Программная инженерия. Тема 1. Теория Программирования. 50 с.; Тема 2. Технология программирования, 48; Тема 3. Базовые основы программной инженерии. — 52с. / Методические пособия, Москва, МФТИ, 2016.
23. Лаврищева Е.М., Петренко А.К. Моделирование систем и их семейств // Труды ИСП РАН. — М.: 2016. — Том 28. — Вып. 6. — С.180–190.
24. Кулямин В.В., Лаврищева Е.М., Мутилин В.С., Петренко А.К. Верификация и анализ переменных операционных систем // Труды ИСП РАН. — М.: 2017. — Том 28. — Вып. 3. — С.189–209.
25. Лаврищева Е.М. Программная инженерия. Парадигмы, Технологии, CASE- средства программирования, 2 изд. — М.: Юрайт, 2016. — 280 с.
26. Лаврищева Е.М. Программная инженерия и технология программирования сложных систем // М.: Юрайт, 2017. — 431с.
27. Лаврищева Е.М., Колесник А.Л., Стеняшин А.Ю. Объектно-компонентное проектирование программных систем // Теоретические и прикладные вопросы. — Вестник КНУ, серия физ.-мат. наук. — Киев, 2013. — №4. — С. 150–164; Ekaterina Lavrischeva, Andrey Stenyashin, Andrii

- Kolesnyk. Object-Component Development of Application and Systems. Theory and Practice // Journal of Software Engineering and Applications. — 2014. — Vol. 7. — No. 9. — doi:10.4236/jsea.2014.79070.
28. Ekaterina M. Lavrischeva. Assembling Paradigms of Programming in Software Engineering // Journal of Software Engineering and Applications. — 2016. — Vol. 9. — No. 6. — P. 296–317. — doi:10.4236/jsea.2016.96021.
  29. Lavrischeva Ekaterina. Ontological Approach to the Formal Specification of the Standard Life Cycle // Science and Information Conference-2015, July 28–30, London, UK, www.conference.thesai.org. — P. 965–972.
  30. MacGregor S.D., Sykes D.A. Practical Guide to Testing Object-oriented software.- Addison-Wesley, 2001.
  31. Sayyad A. S., Ingram J., Menzies T., Ammar H. Scalable product line configuration: a straw to break the camel's back // Proc. of IEEE/ACM 28-th International Conference on Automated Software Engineering (ASE 2013). — IEEE, 2013. — P. 465–474. — doi:10.1109/ASE.2013.6693104.
  32. Лаврищева Е.М., Рыжов А.В. Подход к созданию систем и сайтов из готовых ресурсов // Научный сервис в сети Интернет: труды XX Всероссийской научной конференции (17–22 сентября 2018 г., г. Новороссийск). - М.: ИПМ им. М.В. Келдыша, 2018. — С. 321–346. — ISBN 978-5-98354-046-0.
  33. Е.М. Lavrischeva, А.К. Petrenko. Informatics-70. Computerization aspects of programming software and informatic systems technologies // Proc. ISP RAS. — 2018. — Vol. 1. — Issue 2. — P. 3–4.
  34. Лаврищева Е.М., Аветисян А.И., Петренко А.К. Информатика. ЭВМ-70. Анализ и аспекты развития. — Доклад на конференции ISPRAS-2018. (<http://0x1.tv/20181122AF>).
  35. Е.М. Lavrischeva. The Scientific Basis of Software Engineering // International Journal of Applied and Natural Sciences (IJANS). — Vol. 7. — Issue 5. — Aug–Sep 2018. — P. 15–32. — ISSN(P): 2319-4014; ISSN(E): 2319-4022.
  36. Лаврищева Е.М., Пакулин Н.В., Рыжов А.Г., Зеленов С.В. Анализ методов оценки надежности оборудования и систем. Практика применения методов // Труды ИСП РАН. — Том 30. — Вып. 3. — 2018. — С. 99–120. — doi:10.15514/ISPRAS-2018-30(3)-8. <http://0x1.tv/20180517F>
  37. Лаврищева Е.М. Современные системы искусственного интеллекта // Симпозиум «Искусственный интеллект: различные подходы к его воплощению (компьютерно-сетевые, нейросетевые, квантовые технологии и другие)», Москва, 13 октября 2018. — Федеральное государственное образовательное учреждение при Правительстве РФ. Колледж информатики и программирования. — Доклад <https://videonauka.ru/stati/30-metodika-prepodavaniya-tekhnicheskikh-distiplin/237-informatika-i-evm-70-analiz-i-aspekty-razvitiya>.
  38. Лаврищева Е.М. Теория графового моделирования сложных систем из модулей для прикладных областей.- Научно-практический журнал «Высшая школа, июль (14) 2019.- с.27-46.
  39. Лаврищева Е.М., Петренко А.К. Моделирование операционных, программных и технических систем в проектах РФФИ. Труды Института системного программирования РАН. 2021; 33(6):265-280. [https://doi.org/10.15514/ISPRAS-2021-33\(6\)-19](https://doi.org/10.15514/ISPRAS-2021-33(6)-19)
  40. Лаврищева Е.М., Петров И.Б. Технология моделирования физических и математических задач предметных областей знаний. «Евразийское Научное Объединение», Физико-математические науки, № 1 (71), Январь, 2021.
  41. Лаврищева Е.М., Петренко А.К., Зеленов С.В., Козин С.В. Технология сборочного создания экспериментального варианта ядра OS Linux с обеспечением качества для применения в прикладных и технических системах. Наука и технологии. Системный администратор. Ноябрь, 2021. с.80-89.
  42. Лаврищева Е. Развитие программного обеспечения вычислительной техники в СССР и в рамках межправительственной комиссии (МПК) СССР – СЭВ (1969 – 1991). Пятая международная конференция SoRuCom-20 (6–7 октября 2020 г., Москва), с. 200-208.
  43. Лаврищева Е.М., Петров И.Б. Моделирование технических и математических задач прикладных областей знаний на ЭВМ. Труды Института системного программирования РАН. 2020; 32(6):167-182. [https://doi.org/10.15514/ISPRAS-2020-32\(6\)-13](https://doi.org/10.15514/ISPRAS-2020-32(6)-13)
  44. Лаврищева Е.М., Зеленов С.В. Модельный подход к обеспечению безопасности и надежности Web-сервисов. Труды Института системного программирования РАН. 2020; 32(5):153-166. [https://doi.org/10.15514/ISPRAS-2020-32\(5\)-12](https://doi.org/10.15514/ISPRAS-2020-32(5)-12)
  45. Лаврищева Е.М., Петренко А.К. Технология сборки интеллектуальных и информационных

ресурсов Интернет. Научный сервис в сети Интернет: труды XXI Всероссийской научной конференции (23-28 сентября 2019 г.,Новороссийск). — М.: ИПМ им. М.В. Келдыша, 2019. — С. 469-488. <https://doi.org/10.20948/abrau-2019-93>

46. Лаврищева Е.М., Зеленев С.В., Пакулин Н.В. Методы оценки надежности программных и технических систем. Труды Института системного программирования РАН. 2019; 31(5):95-108. [https://doi.org/10.15514/ISPRAS-2019-31\(5\)-7](https://doi.org/10.15514/ISPRAS-2019-31(5)-7)
47. Лаврищева Е.М., Мутилин В.С., Козин С.В., Рыжов А.Г. Моделирование прикладных и информационных систем из готовых сервисных ресурсов Интернет. Труды Института системного программирования РАН. 2019; 31(1):7-24. [https://doi.org/10.15514/ISPRAS-2019-31\(1\)-1](https://doi.org/10.15514/ISPRAS-2019-31(1)-1)
48. Е. М. Лаврищева, И. Б. Петров, А. К. Петренко. Парадигмы моделирования и программирования задач предметных областей знаний: монография / DirectMeduio. Москва- Берлин, 2021. 495 с.