

DOI: 10.15514/ISPRAS-2020-32(5)-4



Практика и перспективы применения семейства эмуляторов архитектур мейнфреймов IBM

A.V. Shmid, ORCID: 0000-0002-4672-1458 <ashmid@ec-leasing.ru>
 ЗАО «ЕС-лизинг»,

117587, Россия, г. Москва, Варшавское шоссе, д. 125, стр.1

Национальный исследовательский университет «Высшая школа экономики»,
 Россия, 101000, г. Москва, ул. Мясницкая, д. 20

Аннотация. В статье представлено описание семейства эмуляторов архитектур мейнфреймов IBM, история их разработки, их функциональные особенности и возможности применения, а также опыт многолетнего развития (начиная с 1994 года) состава эмуляторов, и сфер их применения. Решена относительно простая по современным представлениям задача создания виртуальной машины в операционной системе VSE/ESA для переноса в эту целевую среду унаследованных платформозависимых приложений. Задача решена сначала для ЕС ЭВМ в РФ, а потом для машин IBM 9221 в ФРГ и других западных странах. Перенос осуществлялся в среду современной по тем временам OS/390, а также IBM AIX. Обеспечена возможность виртуального исполнения любых существующих операционных систем мейнфреймов IBM в средах основных серверных ОС: Linux, Windows, AIX, Z/OS, ZLinux. Разработано решение по объединению образовавшихся виртуальных вычислительных узлов любых типов в гетерогенные территориально-распределенные вычислительные сети, обеспечивающие в частности множественное взаимное резервирование узлов в сети.

Ключевые слова: интерпретативное исполнение; эмулятор; хост система; гостевая система; IBM

Для цитирования: Шмид А.В. Практика и перспективы применения семейства эмуляторов архитектур мейнфреймов IBM. Труды ИСП РАН, том 32, вып. 5, 2020 г., стр. 57-66. DOI: 10.15514/ISPRAS-2020-32(5)-4

Practice and Prospects for Using the Emulator Family of IBM Mainframe Architecture

A.V. Shmid, ORCID: 0000-0002-4672-1458 <ashmid@ec-leasing.ru>
 EC-leasing Co.,

Warszawskoye Shosse, Moscow, 117587, Russia

National Research University Higher School of Economics,
 20, Myasnickaya st., Moscow, 101000, Russia

Abstract. This article describes the family of emulators for IBM mainframe architectures, their development history, functional features and capability, as well as the experience of many years (since 1994) of emulators development and their implementation area. There was solved the relatively simple task (for modern standards) of creating a virtual machine in the VSE/ESA operating system for transferring legacy platform-dependent applications to this target environment. The problem was solved at first for EU computers in Russia, and then for IBM 9221 in Germany and in the other western countries. The transfer was made to the OS/390 environment, and to IBM AIX, quite modern at that time. The virtual execution of any existing IBM mainframe operating systems in the main server OS environments: Linux, Windows, AIX, Z/OS, ZLinux had been provided. There was developed the solution for combining any types of formed virtual computing nodes into heterogeneous geographically distributed computing networks that provide, in particular, multiple mutual redundancy of nodes in the network.

Keywords: interpretive execution; emulator; host system; guest system, IBM

For citation: Shmid A.V. Practice and prospects for using the emulator family IBM mainframe architecture. Trudy ISP RAN/Proc. ISP RAS, vol. 32, issue 5, 2020, pp. 57-66 (in Russian). DOI: 10.15514/ISPRAS-2020-32(5)-4

1. Введение и постановка задачи

В настоящее время мейнфреймы IBM традиционно применяются в ряде критически важных автоматизированных систем управления (АСУ) РФ. Достаточно назвать два критически важных как для населения, так и для экономики примера: ЦБ РФ и Пенсионный фонд.

И эти системы, как и любые другие АСУ на любых других платформах, подвергаются ряду внешних и внутренних угроз. Но именно в силу значения хотя бы этих примеров для экономики и населения РФ любые работы, позволяющие устранить для систем этого класса хотя бы некоторые из существующих угроз, несомненно, актуальны.

Наша история борьбы с различными угрозами для машин этого класса началась очень давно. К середине 90-х годов в РФ был прекращен выпуск вычислительных машин ЕС ЭВМ и перспективы возможности дальнейшей эксплуатации многочисленных унаследованных АСУ ЕС ЭВМ оказались под угрозой.

Одновременно появилась возможность поставки в РФ по приемлемым ценам архитектурно близких ЕС ЭВМ вычислительных машин IBM на то время новейшей архитектуры вместе с операционной системой (ОС) VSE/ESA как платформы, замещающей ЕС ЭВМ.

Проблема заключалась в том, что унаследованные АСУ были платформозависимы от ОС ЕС ЭВМ, и их перенос в среду исполнения ОС IBM потребовал бы значительной непредсказуемой по трудоемкости индивидуальной переработки каждой АСУ по отдельности.

Была поставлена задача: найти универсальное массовое решение, не требующее индивидуальной работы с каждой из АСУ ЕС ЭВМ при ее переносе в среду VSE/ESA. Проще говоря, по постановке задачи требовалось переносить АСУ «как есть и за один день». Наличие такого решения позволило бы сохранить на длительную перспективу огромный программный задел АСУ ЕС ЭВМ.

Стало очевидным, что в силу такой постановки задачи почти единственно возможным решением возникшей проблемы являлся бы перенос АСУ вместе с ее ОС ЕС в виртуальную машину ОС VSE/ESA, эмулирующую для ОС ЕС архитектуру ЕС ЭВМ в полном объеме.

Альтернатива переноса АСУ вместе с ОС ЕС в виртуальную машину ОС VM IBM была отвергнута по ценовым и техническим соображениям.

Эмулятор с заданной функциональностью для VSE/ESA был нами разработан и успешно прошел тестирование в лаборатории IBM в Бейлингене, ФРГ.

Вскоре выяснилось, что подобные проблемы продления сроков эксплуатации платформозависимых унаследованных приложений имелись как в ФРГ, так и в других странах. Так, оказалось, что в результате прекращения выпуска вычислительных машин IBM 9221 их операционная система DPPX с критически важными для ФРГ на последующий длительный период приложениями (полный аналог ситуации РФ) осталась без вычислительных машин, без дисков и без сетевых адаптеров, с которыми DPPX должна работать.

С учетом дополнительных требований и на основе эмулятора для VSE/ESA был разработан эмулятор архитектур S/370, S/390 IBM для IBM OS/390, который позволял исполнять в гостевом режиме все существовавшие на тот момент ОС архитектур IBM (включая DPPX) с необходимым набором эмулируемых (в том числе уже снятых с производства) внешних устройств IBM. На этот эмулятор (ISX – Interpretative Space Executive) в 2003 году нами был получен патент США [1].

Проблема замены стоящих в поле множества машин IBM 9221 на современное оборудование IBM с OS/390 в ФРГ была успешно решена, и этот опыт был транслирован IBM и в другие страны.

Позже, по желанию заказчика (IBM) была также разработана версия подобного эмулятора для операционной системы AIX машин IBM P-series.

В итоге нами было реализовано несколько сотен лицензий эмуляторов для переноса унаследованных приложений вместе со средой их исполнения на платформы AIX и OS/390. Затем были также разработаны версии этого эмулятора для ОС Windows и Linux.

Когда появилась информация о Z-архитектуре IBM, эмуляция S/390 была расширена до уровня Z-архитектуры.

Со временем появилось целое семейство различных эмуляторов, отличительной особенностью которых явилась детальная эмуляция как принципов операций архитектур мейнфреймов от IBM S/370 до IBM Z [2-8], так и принципов работы всех видов внешних и сетевых устройств [9-14] в соответствии с технической документацией фирмы изготовителя [2-8]. Причем эмуляция могла быть исполнена на всех программных платформах основных вендоров.

В результате была обеспечена принципиальная возможность полного, либо частичного быстрого перевода прикладных систем, ранее разработанных для мейнфреймов IBM, на оборудование других вендоров, уже установленное в компании.

Как следствие, появились новые постановки задач по оптимизации таких общесистемных характеристик теперь принципиально гетерогенных систем как совокупная стоимость владения, доступность, катастрофоустойчивость.

2. Семейство эмуляторов ISX – системные возможности

К настоящему времени разработано семейство из 10 версий эмулятора ISX, в совокупности обеспечивающее возможности гостевого исполнения всего ассортимента ОС IBM для мейнфреймов IBM (как ранее выпускавшихся, так и современных – Guest OS на рис. 1) на оборудовании современных архитектур различных вендоров – Real Machine на рис. 1, в средах основных современных ОС – Host OS на рис 1.

Причем с точки зрения системного программирования для любой из Host OS (рис. 1) ISX вместе с исполняемой им гостевой ОС представляет собой единицу работы (Workunit на рис. 1), участвующую в борьбе за ресурсы этой ОС по ее правилам.

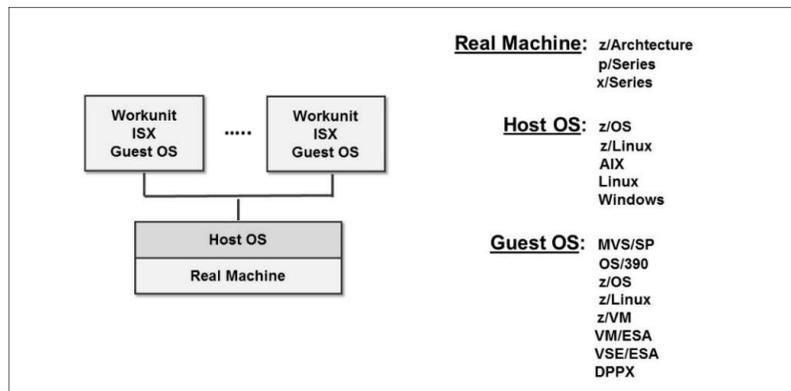


Рис. 1. Программа - исполнитель интерпретативного пространства
Fig. 1. Hardware Interpreter – Interpretive Space eXecutive (ISX)

Также с точки зрения системного программирования, отличия группы эмуляторов семейства, исполняемых на оборудовании z/Architecture, от оборудования прочих архитектур (рис. 1) состоит в том, что в случае машин IBM Z эмуляция ведется с использованием режима интерпретативного исполнения самой архитектуры: команды SIE – Start Interpretative Execution. Поэтому в этом случае производительность эмуляционного решения практически не снижается

В то же время для прочих архитектур в интересах эмулятора прописана вся система команд архитектуры любой из архитектур мейнфреймов IBM. По этой причине на производительность эмулятора ISX на любой платформе, кроме IBM/Z, очевидно влияют и издержки на эмуляцию архитектуры.

Кроме того, во втором случае исполнение гостевой системы практически ведется в контролируемой среде. Эмулятор не принимает к исполнению запросов гостевой ОС на исполнение команд, не прописанных в нем разработчиками эмулятора.

Эмуляторы ISX обеспечивают также эмуляцию как ранее производимого, так и современного оборудования мейнфреймов IBM (процессоров, каналов, устройств ввода-вывода, сетевых устройств машин IBM архитектур S/ 370, S/390, Z – табл. 1), обеспечивающую возможность исполнения Guest OS в средах любых Host OS (в том числе иных вендоров) на оборудовании этих вендоров (рис. 1).

Табл. 1. Эмуляция устройств ввода-вывода
Table 1. I/O Device Emulation

Unit record devices	3215, 2540R, 2540P, 1403
FBA devices	9332, 9335, 9336
CKD devices:	3380, 3390
Magnetic tape devices	3480, 3490
Display units	3274 Models 2, 3, 4, 5
Tokin-Ring adapter	IEEE 802.5
Channel-to-channel adapter	
LCS adapter	8232
OSA Express adapter	

Различные аспекты эмуляции как архитектур, так и внешних устройств общеизвестны и рассмотрены в ряде работ, например, [1], [15-19].

Основной же особенностью рассматриваемого семейства эмуляторов (как следует из рис. 1 и табл. 1) является его функциональная полнота, обеспечивающая перенос приложений с мейнфреймов на платформы большинства основных вендоров.

По мере разрастания семейства эмуляторов по охвату вендорных платформ и приобретения опыта работы с самыми разнообразными клиентами, становилось очевидной необходимостью перехода к решению и более сложной задачи.

А именно, требовалось разработать также и инструменты взаимодействия между собой множества виртуальных вычислительных узлов ISX (в простейшем случае двух узлов: основного и резервного). В общем же случае нужно было поставить и решить технологическую задачу организации сетевого взаимодействия множества виртуальных вычислительных территориально-распределенных узлов ISX, работающих на платформах различных вендоров.

Эта технологическая задача также была решена, что и предопределило направление дальнейшего развития проекта и его перспективы.

3. Перспективы: от локальных резервированных ЦОД к построению гетерогенных территориально-распределенных резервированных вычислительных сред узлов ISX

В этом разделе перечисляются новые системные возможности семейства эмуляторов ISX, в совокупности обеспечивающие возможность построения гетерогенных территориально-распределенных резервированных вычислительных сред узлов ISX. Приводятся примеры их применения.

3.1 Системные возможности эмуляторов ISX и команды оператора по управлению узлами ISX

Концептуально управление множеством узлов ISX производится операторами узлов с применением команд управления узлом, доступных оператору.

Команды оператора позволяют задать окружение гостевой операционной системы (количество процессоров, размер памяти, используемые устройства ввода-вывода и т.д.). Гостевая система может быть загружена для исполнения командой IPL.

ISX обеспечивает команды для эмуляции аппаратных прерываний в гостевую систему, команды останова и возобновления работы гостевой системы, команды наблюдения за работой гостевой системы, команды снятия и анализа дампов гостевой системы, команды управления миграцией гостевых систем, команды обслуживания копий дисков гостевых систем. В табл. 2 представлен полный список команд оператора по управлению эмулятором ISX.

Табл. 2. Команды ISX

Table 2. ISX commands

BEGIN	продолжить исполнение гостевой системы
CAC	ввести данные для гостевой консоли
CACHE	сохранить историю исполнения гостевых инструкций
DEFINE	определить устройство ввода-вывода для гостевой системы
DETACH	удалить устройство ввода-вывода из гостевой конфигурации
DISPLAY	показать данные гостевой системы
DUMP	сохранить состояние гостевой системы
EXEC	выполнить последовательность команд эмулятора из командного файла
EXTERNAL	сгенерировать прерывание от ключа внешних прерываний
HELP	получить информацию о командах эмулятора
HMC	ввести данные для пультовой консоли
LOG	определить ведение журнала входных/выходных сообщений
IPL	выполнить начальную загрузку гостевой системы
PAUSE	приостановить процесс исполнения команд эмулятора
RECEIVE	принять данные состояния гостевой системы
RESTART	сгенерировать рестарт-прерывание
SHOW	показать состояние гостевой системы
SEND	послать данные состояния гостевой системы
SET	установить параметры работы эмулятора

SHUTDOWN	завершить работу эмулятора
STOP	остановить исполнение гостевой системы
STORE	изменить данные гостевой системы
SYNCH	синхронизировать базу и копию гостевого диска
TAKE	переопределить параметры устройства ввода-вывода
TRACE	получить трассу событий в гостевой системе

3.2 Локальные и удаленные диски гостевых систем

ISX поддерживает локальные и удаленные CKD DASD устройства. При определении дисков для гостевых систем оператором можно указать, в каком узле сети – собственном или удаленном – располагаются данные диска. Гостевая система «не замечает» разницы между локальным и удаленным диском, кроме времени исполнения операций ввода/вывода

Для работы с удаленным диском могут использоваться механизмы сжатия данных при пересылке данных по сети и буферизации в памяти для уменьшения нагрузки на сеть и ускорения операций ввода-вывода.

3.3 Ведение копий дисков гостевых систем

ISX поддерживает ведение локальных и удаленных копий CKD DASD устройств. Если для диска (базы) определена копия, то данные диска будут прочитываться из базы, пока она работоспособна, и будут записываться и в базу, и в копию. Если база перестает быть работоспособной, то копия начинает исполнять роль базы.

Сигнал о завершении операции ввода-вывода, полученный от узла ISX, в котором размещена копия, подтверждает синхронизацию дорожки диска. Если копия теряет связь с эмулятором ISX, который исполняет гостевую систему, то поддерживается только базовый том. Если базовый том теряет связь с эмулятором ISX, который исполняет гостевую систему, то поддерживается только копия. При восстановлении связи копия и база автоматически синхронизируются.

Первоначально копии дисков создаются при помощи команды оператора ISX SYNCH. Эта команда включает процесс создания копии диска независимо от того, исполняется ли гостевая система или нет.

3.4 Миграция гостевых систем по узлам ISX

ISX поддерживает специальные возможности для передачи образа гостевой системы в удаленный узел в сети TCP/IP. Для этого используются команды оператора SEND и RECEIVE. Если дисковые устройства гостевой системы имеют копии в удаленном узле, может быть разработан сценарий полной миграции гостевой системы из одного узла в другой и обратно или в третий узел. Пересылаются состояние процессоров, памяти, устройств ввода-вывода на момент исполнения команды SEND по указываемому в команде сетевому адресу. Гостевая система в текущем узле прекращает свою работу и возобновляет свою работу в новом узле сети TCP/IP после исполнения команды RECEIVE в этом узле.

3.5 Характерные примеры построения сетей гостевых ОС

На рис. 2 приведена типичная схема реализации критически важных приложений на мейнфрейме IBM для большой организации.

Как правило, на мейнфрейме заводится множество (несколько десятков) логических разделов (LPAR), в которых размещаются операционные системы Z/OS и Z/Linux.

На рис. 2 во всех LPAR все ОС находятся в гостевом режиме под ISX, что практически не влияет на производительность, поскольку виртуализация для гостевой ОС осуществляется интерпретативными командами самого мейнфрейма.

При этом, однако, виртуализация узла исполнения под ISX на основной машине Z придает этому узлу новые возможности сетевого взаимодействия с другими виртуальными узлами, в том числе на платформах любых других вендоров.

Например, новые возможности (по командам операторов виртуальных узлов): такие, например, как заведение любого количества виртуальных дисков в удаленных узлах, ведение синхронных копий, перевод на обработку образа z/OS в любую другую географическую точку, в том числе на платформу другого вендора.

Поскольку виртуальные внешние по отношению к z Machine узлы (рис. 2) могут находиться удаленно в различных географических точках, то может быть организовано множественное распределенное синхронное резервирование системы в целом или отдельно по каждому из LPAR, показанных на рис. 2, в том числе на виртуальных узлах другого удаленно расположенного мейнфрейма.

Таким образом, это решение может быть рассмотрено как импортозамещающая технология множественного территориально-распределенного резервирования критически важных приложений для мейнфреймов.

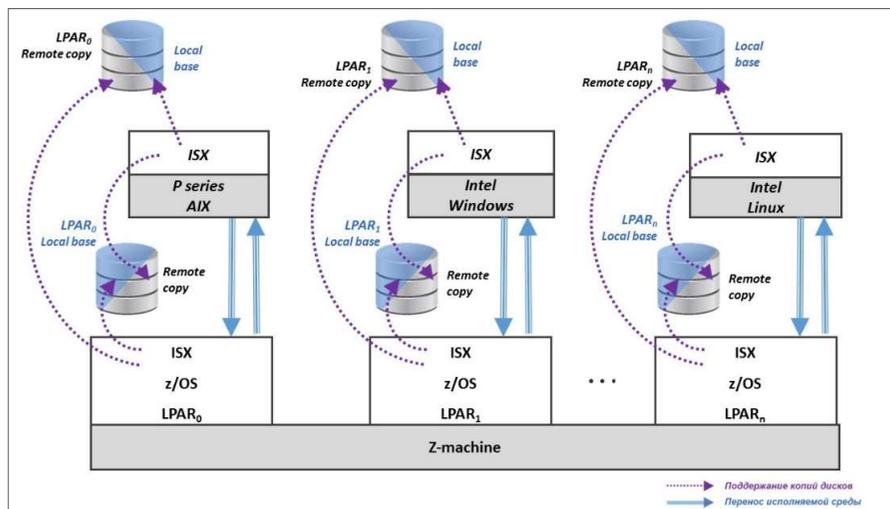


Рис. 2. Схема реализации критически важных приложений на мейнфрейме
Fig. 2. Implementation diagram for critical applications on the mainframe

Наконец, на рис. 3 показано, что в силу полной эмуляции всех архитектур IBM мейнфрейм для всех HOST машин, за исключением самой IBM Z, появление в будущем машин любой неизвестной архитектуры, работающей под управлением Linux, позволит безболезненно включить это вновь появившееся оборудование в проекты с предложенной технологией. Можно условно квалифицировать эту структуру как гетерогенную территориально-распределенную вычислительную сеть, в которой наращивание мощностей и функциональности, любые ремонты принципиально могут вестись без прерывания работы сети в целом – временным переводом обработки гостевой ОС со всеми приложениями с модифицируемого либо ремонтируемого узла в любой другой узел по команде оператора.

Также без остановки сети в целом динамически могут быть проведены изменения схемы резервирования узлов.

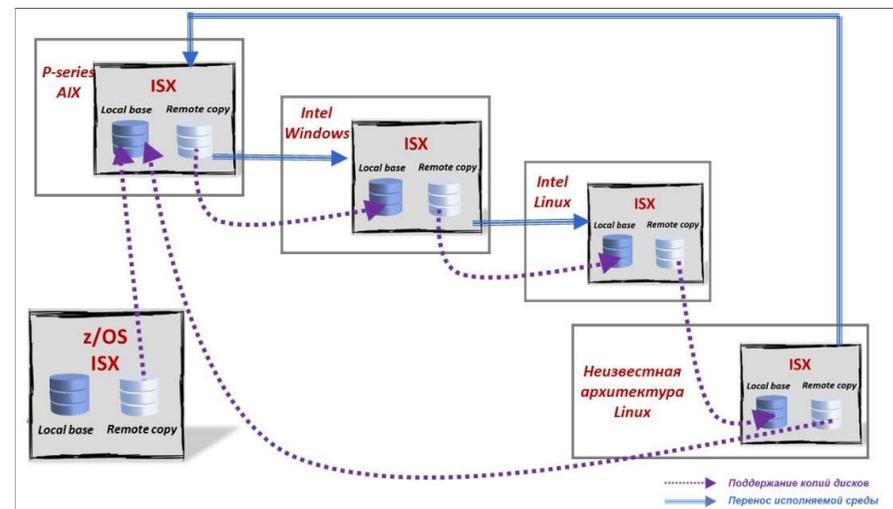


Рис. 3. Пример гетерогенной территориально-распределенной вычислительной среды
Fig. 3. Sample of a heterogeneous geographically distributed computing environment

Проблемы проектирования гетерогенных сетей рассмотрены в множестве работ, например, [20-27].

Концептуально наиболее близкое к ISX решение, обеспечивающее гостевое исполнение OS zLinux (только) в среде zLinux Z машин, а также в средах некоторых других платформ, описано в [28].

Однако присущие ему ограничения как по HOST, так и по GUEST OS не обеспечивают как решения задачи по переносу унаследованных приложений, так в общем случае и задачи по построению гетерогенных сетей в связи с отсутствием ряда необходимых для таких решений функциональных возможностей, например, функционально полного синхронного ведения копий дисков (см. табл. 2).

Особенностью же предлагаемого технического решения ISX, как семейства взаимодействующих эмуляторов, является тот факт, что его можно рассматривать в качестве «конструктора» с функциональной полнотой, необходимой и достаточной для самостоятельного создания набора виртуальных узлов ISX на платформах большинства вендоров, последующего объединения виртуальных узлов ISX в гетерогенную территориально-распределенную вычислительную среду, управляемую операторами среды с применением уже существующих инструментов управления созданными виртуальными узлами ISX (табл. 2).

4. Выводы

1. Разработанная технология переноса унаследованных приложений архитектур S/370, S/390, Z (мейнфреймов), допускающая их перенос на любую из ныне существующих серверных платформ, обеспечивает долгосрочную стратегическую устойчивость этих (критически важных) приложений РФ, в том числе в условиях импортозамещения.
2. Представляемая технология реализует альтернативный вариант множественного территориально-распределенного резервирования критически важных систем на

полностью отечественных программных продуктах, повышая живучесть таких проектов с перспективой простого переноса уже существующих проектов на полностью отечественное оборудование по достижению этим оборудованием целевой для проекта производительности.

В случае ее применения возникает также возможность динамического перемещения обработки оператором (пока) в наименее загруженные узлы вычислительной сети, несомненно приводящая к снижению совокупной стоимости владения проекта в целом.

3. Можно также констатировать, что проблемы переноса платформозависимых старых унаследованных приложений названных архитектур на какую-либо платформу без каких-либо изменений все-таки существуют и поныне.

Только в РФ можно назвать более 100 крупных предприятий, которые сталкиваются с подобными проблемами.

Список литературы/References

- [1] Alexander V. Shmid, Viacheslav V. Naumov. Virtual machines in OS/390 for execution of any guest system. US 6,530,078 B1 USA, 4 Mart 2003.
- [2] GA22-7000-10. IBM System/370. Principles of Operation. 1987.
- [3] SA22-7201-04. IBM Enterprise System Architecture/390. Principles of operation. 1997.
- [4] SA22-7832-07. IBM z/Architecture. Principles of Operation. 2009.
- [5] SA22-7095-1. IBM System/370 Extended Architecture Interpretive Execution. 1985.
- [6] GA26-1660-1. IBM3310 Direct Access Storage Reference Manual.
- [7] GA32-0274-05. IBM 3990/9390 Storage Control Reference.
- [8] GA32-0127-03. IBM 3490 Magnetic Tape Subsystem Hardware Reference.
- [9] GA23-0059-07. 3270 Information Display System. Data Stream Programmer's Reference.
- [10] SA24-4236-03. Enterprise System/9000 Models 120,130,150,170, and 200. Work Station Subsystem Description and Reference.
- [11] SA33-1600-03. Enterprise System/9000. 9221 processors. IBM Token-Ring and IEEE802.3 Local Area Network. Programming Information.
- [12] SA22-7203-00. Enterprise Systems Architecture/390. ESCON Channel-to_Channel Adapter.
- [13] SA22-T091-01. Channel-to-Channel Adapter for the System/360 and System/370 I/O Interface.
- [14] GC23-3870-01. IBM System/390. Planning for the System/390. Open Systems Adap.
- [15] S. Matic. Emulation of hypercube architecture on nearest-neighbor mesh-connected processing elements. IEEE Transactions on Computers, vol. 39, № 5, 1990, pp. 698-700.
- [16] M. Guttenbrunner, A. Rauber. Evaluating Emulation and Migration: Birds of a Feather? Lecture Notes in Computer Science, vol. 7634, 2012, pp. 158-167.
- [17] D. Seto, M. Watanabe. A dynamic optically reconfigurable gate array – perfect emulation. IEEE Journal of Quantum Electronics, vol. 44, № 5, 2008, pp. 493-500.
- [18] R. Helaihel, K. Olukotun. Emulation and prototyping of digital systems. NATO ASI Series, vol. 310, Hardware/Software co-design, 1996, pp. 339-366.
- [19] A. Suzuki, S Oikawa. Implementing a simple trap and emulate VMM for the ARM architecture. In Proc. of the 17th International Conference on Embedded and Real-Time Computing Systems and Applications, vol. 1, 2011, pp. 371-379.
- [20] M. Fang et al. High Performance X86 Emulation IO Architecture on Heterogeneous Processor Platform. In Proc. of the International Conference on Computational and Information Sciences, 2010, pp. 944-947.
- [21] R.W. Heath, M. Kountouris. Modeling heterogeneous network interference. In Proc. of the Information Theory and Applications Workshop, 2012, pp. 17-22.
- [22] B.H. Jung, N.O. Song, D.K. Sung. A network-assisted user-centric WiFi-offloading model for maximizing per-user throughput in a heterogeneous network. IEEE Transactions on Vehicular Technology, vol. 63, № 4, 2013, pp. 1940-1945.
- [23] B. Ahlgren et al. Ambient networks: bridging heterogeneous network domains. In Proc. of the 16th International Symposium on Personal, Indoor and Mobile Radio Communication, vol. 2, 2005, pp. 937-941.
- [24] Q. Li et al. Intracell cooperation and resource allocation in a heterogeneous network with relays. IEEE Transactions on Vehicular Technology, vol. 62, № 4, 2012, pp. 1770-1784.

- [25] J. Liu et al. Uplink power control and interference coordination for heterogeneous network. In Proc. of the 23rd International Symposium on Personal, Indoor and Mobile Radio Communications, 2012, pp. 519-523.
- [26] T. Alpcan, J.P. Singh, T. Başar. Robust rate control for heterogeneous network access in multihomed environments. IEEE Transactions on Mobile Computing, vol. 8, № 1, 2008, pp. 41-51.
- [27] D. Kutscher, J. Ott. Service maps for heterogeneous network environments. In Proc. of the 7th International Conference on Mobile Data Management (MDM'06), 2006, 27 p.
- [28] QEMU s390x Guest Support. URL: <https://wiki.qemu.org/Documentation/Platforms/S390X>, accessed 11.11.2020.

Информация об авторе / Information about author

Александр Викторович ШМИД – д.т.н., проф., генеральный директор ЗАО «ЕС-лизинг», заведующий базовой кафедрой ЗАО «ЕС-лизинг» «Информационно-аналитические системы» МИЭМ НИУ ВШЭ. Сфера научных интересов: big data.

Alexander Viktorovich SHMID – Doctor of Technical Sciences, Prof., Chief Executive Officer «EC-leasing» Co, Head of the Department of Information and Analytical Systems: Joint Department with EC-leasing, HSE. Research interests: big data.