

## **Віртуальны музей аперацыйных сістэм: арганізацыя мабільнасці і эканоміі рэсурсаў**

*Касцюк Д.А., Луцюк П.А., Уласенка С.С., Жалудок В.А.*

*Брэсцкі дзяржаўны тэхнічны ўніверсітэт, [dmitrykostiuk@bstu.by](mailto:dmitrykostiuk@bstu.by)*

The approach of using virtual machines instead of screenshots in visual timeline of GUI, developed in previous works, is briefly discussed. Mobility of QEMU-based virtualized images is considered as far as problems of their usage. An architecture is presented to minimize CPU usage by keeping unused virtual machines in paused stage.

Адпраўным пунктам для стварэння гэтага праекта [1] стаў апрабаваны раней падыход да ўкаранення віртуальных машын (ВМ) у навучальныя матэрыялы ў ролі інтэрактыўных ілюстрацый, якія замяняюць растравыя выявы з копіямі экранаў і ўстаўныя відэа-ролікі [2]. Тэхнічная інфраструктура, якая дазваляе рэалізаваць такія інфармацыйныя матэрыялы, уключае віртуальныя машыны QEMU, VNC-кліент, напісаны на JavaScript, і HTML5-фрэймворк для адлюстравання інфармацыйных матэрыялаў.

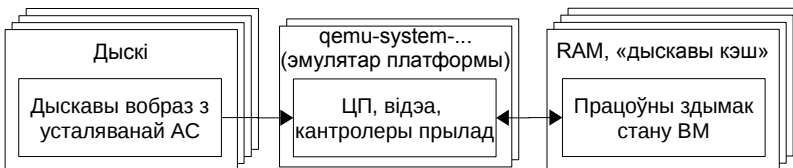
Распрацоўка выкарыстоўвалася спачатку ў якасці лекцыйных матэрыялаў па гісторыі графічнага інтэрфейсу, а затым у ролі дзеючай электроннай выставы: гэта дазволіла выкарыстоўваць інтэрактыўнасць ілюстрацый для магчымасці самастойнага даследавання студэнтамі гістарычных GUI. Экспазіцыя ўяўляе сабой камплект звязаных храналагічна HTML-дакументаў (таймлайн), кожны з якіх змяшчае апісанне асабліва сцяў канкрэтнай графічнай аперацыйнай сістэмы і яе жывую ілюстрацыю ў выглядзе ўбудаванага фрэйма з экранам ВМ. Тое, што распрацоўка грунтуецца на лекцыйным матэрыяле, вызначыла жаданы склад інфармацыйнага кантэнту, які ўключаў два таймлайна: адзін з 40 настольных аперацыйных сістэм і графічных абалонак, а другі — з 30 мабільных АС [2, 3]. Пазней дадаўся трэці таймлайн, які ілюструе развіццё віджэт-тулкітаў.

Зыходная архітэктур мяржавала адначасовую працу ўсіх ВМ (інакш, з-за вялікага аб'ёму дыскавых вобразаў, запуск і прыпынак эмулятараў пры пераходзе паміж старонкамі прывялі б да непажаданых затрымак, асабліва на сістэмах, не аснашчаных цвёрдацельнымі дыскамі). Сярод праблемных сістэмных патрабаванняў такога падыходу — вылічальная магутнасць працэсара і аб'ём RAM, прычым апошні мы лічылі найбольш крытычным. Сапраўды, параўнальна новыя АС выкарыстоўваюць каманды прастаю працэсара, а таму сумяшчэнне вялікага ліку АС, якія праводзяць значную частку часу ў цыкле чакання, не патрабуе вялікай вылічальнай магутнасці. Для зніжэння расходу RAM меркавалася выкарыстоўваць на хост-сістэме тэхналогію kernel samepage merging (KSM), якая забяспечвае эканомію за кошт аднаразовага захоўвання аднолькавых старонак віртуальнай памяці.

Аднак пры набліжэнні колькасці ВМ ў таймлайне да 30 стала відавочным адрозненне рэальнай карціны спажывання рэсурсаў ад чаканай. Агульны аб'ём RAM, заняты адным таймлайнам, склаў менш за 6 Gb без выкарыстання KSM, што дазволіла камфортна запускарць таймлайн на досыць распаўсюджаных для 64-бітных кампутараў 8-гігабайтных канфігурацыях RAM. Разам з тым, тыповая карціна загрузкі працэсара тымі гістарычнымі АС, якія ні ў якім выглядзе не падтрымліваюць каманды CPU idle, прывяла да адчувальнага зніжэння інтэрактыўнасці хост-сістэмы (у ролі тэставых выступалі кампутары з 6-ядзернымі працэсарамі AMD). У выніку была распрацавана новая архітэктара ostimeline версіі 4, с пастаноўкай на паўзу тых ВМ, якія не дэманструюць ў дадзены момант часу.

Была таксама выканана перапрацоўка схемы выкарыстання імгненых здымкаў стану ВМ, якія захоўваюць эмулятарам QEMU ў дыскавых вобразах, згодна тэхналогіі Copy-On-Write. Сутнасць дадзенага прынцыпу заключаецца ў тым, што пры капіраванні файла або яго фрагмента, рэальная копія ствараецца толькі ў той момант, калі АС звяртаецца да скапіраваных байтаў з мэтай запісу ў іх. Гэта дазваляе падзяліць дыскавы вобраз ВМ на дзве часткі: базавы вобраз АС, доступ да якога ажыццяўляецца толькі для чытання, і часовы вобраз, які захоўвае змены, зробленыя ў ходзе работы ВМ. Пры назіранні за асаблівасцямі выкарыстання QEMU дыскавых вобразаў выявілася, што пад час выкарыстання дыскавы файл з базавым вобразам і імгненным здымкам пастаянна папаўняецца невыкарыстоўваемымі зменамі. Па меры збірання ў вобразе такіх зменаў (нябачных для АС з-за імгненнага здымка), шэраг ВМ пачаў дэманстраваць прыкметныя затрымкі пры старце, звязаныя менавіта з аднаўленнем з імгненнага здымка. Эфект не дэманстраваў карэляцыі з памерамі імгненых здымкаў і відавочна звязаны з асаблівасцямі выкарыстання апаратуры пэўнай АС: так, затрымкі дэманстравалі IBM OS / 2 і, нечакана, абалонка CDE на базе live-CD дыстрыбутыва Debian.

У выніку было прынята рашэнне адмовіцца ад сумяшчэння вобразаў устаноўленых АС і імгненых здымкаў у агульным файле на карысць схемы, паказанай на малюнку 1.



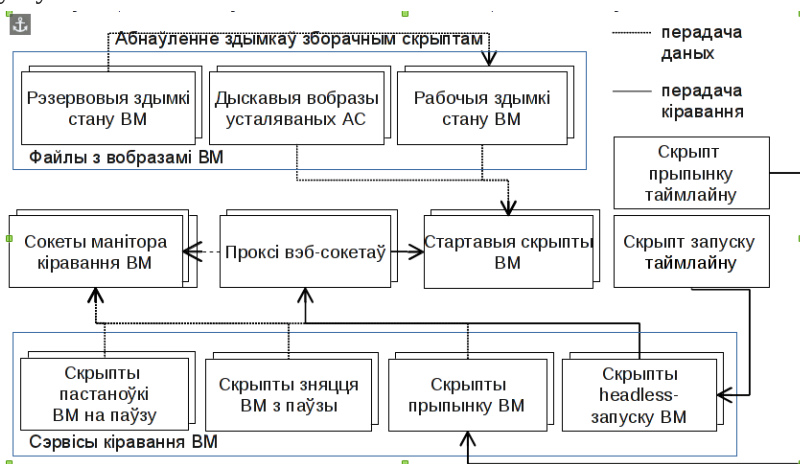
Малюнак 1. Раздзяленне дыскавых вобразаў

Дыскавы вобраз з усталяванай і гадовай да запуску АС з'яўляецца нязменным. У якасці дадатку да базавага ствараецца рабочы («вытворны» у

тэрмінах QEMU) вобраз са здымкам RAM, а таксама зменамі на дыску, якія адбываліся ў працэсе загрузкі АС (умоўна гэта можна назваць «дыскавым кэшам»).

Функцыянаванне ўсёй сістэмы можна бачыць на малюнку 2.

Як можна бачыць з малюнка, файлы з рабочымі здымкамі стану АС захоўваюцца ў двух экзэмплярах: нязменная рэзервовая копія і здымак, які выкарыстоўваецца ў таймлайне. Зборачны скрыпт праглядае падкаталогі, знаходзіць у іх падрыхтаваныя віртуальныя машыны і збірае іх у адзін агульны таймлайн (як згадана ў [3, 4], такое рашэнне выклікана неабходнасцю вар'іраваць прадстаўленыя камерцыйныя АС мінулых гадоў у залежнасці ад наяўных у карыстальніка ліцэнзій). Гэты ж зборачны скрыпт абнаўляе імгненныя здымкі з рэзервовых копій, знішчаючы рост латэнцнасці запуску.



Малюнак 2. Архітэктурна сістэмы

Дыскавыя вобразы і працоўныя здымкі ВМ чытаюцца стартавымі скрыптамі для запуску ВМ (адпаведна, на кожную ВМ прыпадае адзін стартавы скрыпт). Пры непасрэдным запуску стартавы скрыпт запускаяе ВМ ў асобным акне, з адлюстраваннем праз бібліятэку SDL, дзякуючы чаму можна працаваць з кожнай АС непасрэдна, а не ў таймлайне. Пры гэтым выкарыстоўваецца не агульная сістэмная, а лакальная версія QEMU — з-за таго, што сумяшчальнасць паміж імгненнымі здымкамі не заўсёды захоўваецца пры змене версіі QEMU (дакументацыя абяцае захаванне сумяшчальнасці калі пры запуску QEMU дакладна ўказваецца версія, пад якую зроблены здымак, але за апошнія 2 гады гэтае правіла было выпадкова парушана распрацоўшчыкамі QEMU мінімум двойчы).

Пры запуску ў таймлайне ВМ працуе ў headless-рэжыме па выдаленым доступе праз VNC, для чаго да яе стартавага скрыпту перадаюцца дадатковыя

параметры запуску: нумар VNC-порта і файлавы сокет, у які павінен быць перанакіраваны абмен з маніторам QEMU. Такі запуск стартавага скрыпту ажыццяўляецца утылітай `websockify`, якая выконвае заадно перадачу VNC-трафіку ў вэб-сокеты, каб зрабіць даступным гэты трафік VNC-кліенту на JavaScript.

Перанакіраванне ў сокет манітора QEMU выконваецца з дапамогай утыліты `socat` і неабходна для адпраўкі каманд VM (завяршэнне VM, пастаноўка VM на паўзу і зняцце з паўзы). Скрыпты, якія гэта робяць (сэрвісы кіравання VM), ствараюцца аўтаматычна зборачным скрыптам для ўсіх выяўленых VM. Скрыпты прыпынення VM і зняцця з паўзы выконваюцца пры перагортванні старонак у таймлайне, а для праслухоўвання падзей пры перагортванні старонак быў рэалізаваны прымітыўны вэб-сервер, які таксама выкарыстоўвае `socat`.

У выніку выкарыстання прадстаўленай архітэктury загрузка працэсара вызначаецца VM, убудаванымі ў старонку, якая адлюстроўваецца на экране ў дадзены момант, і гэта робіць працу камфортнай на тыповым офісным кампутары.

### ***Спіс літаратуры***

1. Interactive timeline of GUI with live demos of virtualized operating systems used instead of screenshots. <https://gitssh.com/fiowro/ostimeline/>
2. Костюк Д.А. Особенности использования виртуализованных окружений, внедренных в презентационные материалы // Восьмая конференция «Свободное программное обеспечение высшей школе»: тез. докл. / Переславль, 26–27 января 2013 года. М.: Альт Линукс, 2013. – С. 83–86.
3. Касцюк Д.А., Луцко П.А., Уласенка С.С., Жалудок В.А. Ужыванне віртуальных машын у складзе ілюстраваных аглядаў гісторыі праграмага забеспячэння // Третья міжнародна навукова-практычна конференція FOSS Lviv 2014: Збірник навуковых праць / Львів, 24-27 квітня 2014 р. – С. 51 – 54.
4. Kostiuk D., Lutsiuk P., Vlasenko S., Zheludok V. Virtualization-based illustrated reviews of the software history // Открытые технологии: сборник материалов Десятой Международной конференции разработчиков и пользователей свободного программного обеспечения Linux Vacation / Eastern Europe 2014, Гродно, 22 – 24 августа 2014 г. – Брест: Альтернатива. – С. 98 – 101.