Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2011) 1–6

Procedia Computer Science

International Conference on Computational Science, ICCS 2011

SHARE: a web portal for creating and sharing executable research papers

Pieter Van Gorp[a], Steffen Mazanek[b]

[a]p.m.e.v.gorp@tue.nl, Eindhoven University of Technology, The Netherlands
[b]steffen.mazanek@gmail.com, Munich, Germany

Abstract

This extended abstract describes how SHARE (Sharing Hosted Autonomous Research Environments) satisfies the criteria of the Elsevier 2011 Executable Paper Grand Challenge. SHARE is a web portal that enables academics to create, share, and access remote virtual machines that can be cited from research papers. By deploying in SHARE a copy of the required operating system as well as all the relevant software and data, authors can make a conventional paper fully reproducible and interactive. Shared virtual machines can also contain the original paper text — maybe even with embedded computations. In this extended abstract, we outline the workflow that we have leveraged to integrate SHARE successfully in the publication workflow of a journal special issue and various workshop proceedings. We also explain how SHARE is domain independent and how its architecture supports among others the challenge's licensing and scalability requirements.

Keywords: reproducible research, virtualization, web portal, challenge, executable paper, research 2.0

1. Introduction

SHARE [1] has emerged from the organization of the Transformation Tool Contest (TTC, formerly known as GraBaTs), a yearly event aimed at the evaluation and dissemination of advanced transformation techniques and related software. Since TTC is a research contest, it attracts many submissions that rely on software that is still in the prototype phase. This implies, among others, that

1. the software is sometimes not yet publicly released,
2. the software is often difficult to install or configure for proper use with particular inputs,
3. the software is often incomplete or only working in combination with other software, which in turn may require a separate download, installation and license.

In other scenarios, one is struggling with license issues of the data sets that have been used to come to a particular conclusion. Many papers also rely on very large data sets that, irregardless of license issues, are too tedious to download as part of a paper reviewing task or when performing a literature survey and questioning the validity of the alleged research results.

In all of these cases, it would be very convenient if one could simply click a hyperlink within a research paper to arrive at an environment where all software and data related to the paper would be optimally installed and ready for (temporary and secure) evaluation. Since 2009, we provide SHARE as a free academic service for simplifying as much as possible the workflow for creating executable papers.

2. Perspective of the Readers of an Executable Paper in SHARE

This section introduces the SHARE system from the perspective of its most casual user, that is from the perspective of the *reader* of an executable paper (reviewers or others).

Figure 1 shows a usage scenario that is typical for readers of a SHARE-supported publication. Via the browser shown at bullet 1, the reader follows a link from a reference in a (conventional) article. This link points to a webpage, where a specific virtual machine image can be instantiated. Assuming that the reader has never used SHARE before, he first follows a registration procedure (shown at bullet 2). Existing users would simply log in, or would jump to the screen from bullet 3 in case they were already logged in. On the screen shown at bullet 3, the user should just click *Request Session* if he wishes to instantiate the hyperlinked virtual machine image immediately. The SHARE website then balances the load between all virtual machine servers that host the requested virtual machine image. Moreover, it enables users to reserve a future timeslot if all virtual machine servers are fully loaded (see field set "*When?*" on the page from bullet 3).

Figure 1, bullet 4, displays SHARE's main page. In the middle of the page, the details of active sessions are listed. This involves (1) the physical machine at which the virtual machine is running and (2) the port on this server where the Remote Desktop Protocol (RDP) server is listening. In this example, the user has one active session on port *6977* of the machine *jobs.cmi.ua.ac.be*. Bullet 5 shows how the user should enter that information in an RDP client. Users should authenticate using their credentials from the SHARE website. Obviously, the last step (shown at bullet 6) involves working remotely on the virtual machine. To emphasize that users can work concurrently on multiple virtual machines, bullet 6 shows three active RDP sessions. RDP clients are available for most modern operating systems (among others Windows, Linux and Mac). Note that SHARE thus supports multiple operating systems both at the level of the remote virtual machines as well as at the level of the connecting clients.

3. Perspective of Volume Editors

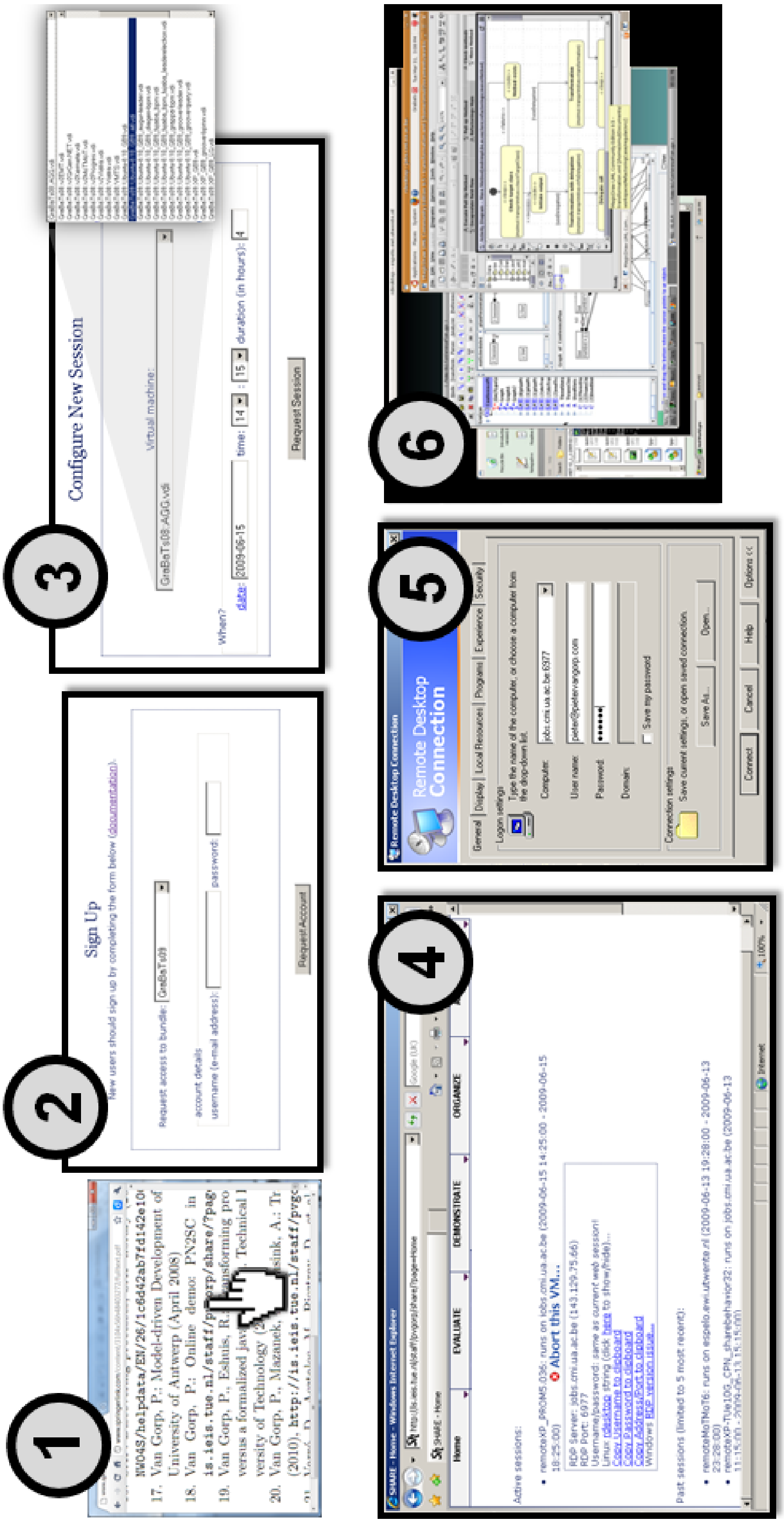
In SHARE, each virtual machine is part of a so-called bundle. Typically, a SHARE bundle relates to a workshop or to a journal issue. Users can subscribe to multiple bundles in order to access the respective machines. Any SHARE user also can apply for *bundle organization* rights. As for other administrative workflows, this would involve submitting a simple form, after which an automated e-mail would be sent to the SHARE users that have the appropriate rights for authorizing the request. For this particular workflow, so-called *bundle administrators* would be notified [1]. Note that SHARE's automated e-mails contain prepared links that minimize the administrative workload.

In most cases, volume editors want to advertize their (executable) papers as much as possible. SHARE not only provides HTML and BibTeX code that can be conveniently adopted in this context, but also provides index pages that enable anonymous visitors of the SHARE website to browse through the list of available virtual machine images. Bundles that are of no interest to the general public can be hidden by their organizers.

4. Perspective of the Authors of Executable Papers

Authors can create new SHARE images based on existing ones by means of a simple "clone" operation. SHARE ensures that clones are only created after approval of both the bundle organizer and the owner of the original image. In many cases, authors simply clone one of the base images, as prepared by the bundle organizer. In other cases, authors re-use images that they (or other authors) have previously contributed to SHARE. The latter often saves precious time in practice.

Figure 2 visualizes the typical flow for the author of an executable paper in SHARE: in step 1, he selects the image he wants to clone, in step 2, the bundle organizer as well as the image owner are notified of this request. Step 3 shows that these stakeholders can decide to postpone the handling of individual requests and handle them in batch via the SHARE website. In this example, we assume that both stakeholders approve the request. In step 4, the author (labeled as *demonstrator d* in the figure) installs any software and data he wishes to share. As displayed by the red crosses, other group members cannot yet launch virtual machines while the image is under preparation. Bullet 5 shows a fragment from the author's view in SHARE. This view provides an overview of all images to which the author has so-called *mutable* (and private) access. As shown in the figure, the author can decide to finalize the image by publishing it as an *immutable* image. Thereafter, it is visible to the peer group members. Alternatively, the image can be discarded



NWO4S/helpdata/EN/26/1c6d42ab7fd142e10
17. Van Gorp, P.: Model-driven Development of
University of Antwerp (April 2008)
18. Van Gorp, P.: Online demo: PN2SC in
is.ieis.tue.nl/staff/pvgorp/share/?pag
19. Van Gorp, P., Eshuis, R.: Transforming pro
versus a formalized java ... Technical
versity of Technology (2
20. Van Gorp, P., Mazanek, ... sink, A.: Tr
(2010), http://is.ieis.tue.nl/staff/pvgc

Sign Up

New users should sign up by completing the form below (documentation).

Request access to bundle: GraBaTs09

account details

username (e-mail address): password:

Request Account

Configure New Session

Virtual machine: GraBaTs09-AGG.vdi

When? date: 2009-06-15 time: 14 : 15 duration (in hours): 4

Request Session

SHARE - Home - Windows Internet Explorer

http://is.ieis.tue.nl/staff/pvgorp/share/?page=Home

Home | EVALUATE | DEMONSTRATE | ORGANIZE

Active sessions:

- remoteXP_PROMS.036: runs on jobs.cmi.ua.ac.be (2009-06-15 14:25:00 - 2009-06-15 18:25:00) Abort this VM...

RDP Server: jobs.cmi.ua.ac.be (143.129.75.66)
RDP Port: 6977
Username/password: same as current web session!
Linux: rdesktop string (click here to show/hide)...
Copy Username to clipboard
Copy Password to clipboard
Copy Address/Port to clipboard
Windows: RDP version issue...

Past sessions (limited to 5 most recent):

- remoteMoTMoT6: runs on espelo.ewi.utwente.nl (2009-06-13 19:28:00 - 2009-06-13 23:28:00)
- remoteXP-Tue10G_CPN_sharebehavior32: runs on jobs.cmi.ua.ac.be (2009-06-13 11:15:00 - 2009-06-13 15:15:00)

Remote Desktop Connection

General | Display | Local Resources | Programs | Experience | Security

Logon settings

Type the name of the computer, or choose a computer from the drop-down list.

Computer: jobs.cmi.ua.ac.be:6977
User name: pieter@pietervangorp.com
Password: ••••••
Domain:

□ Save my password

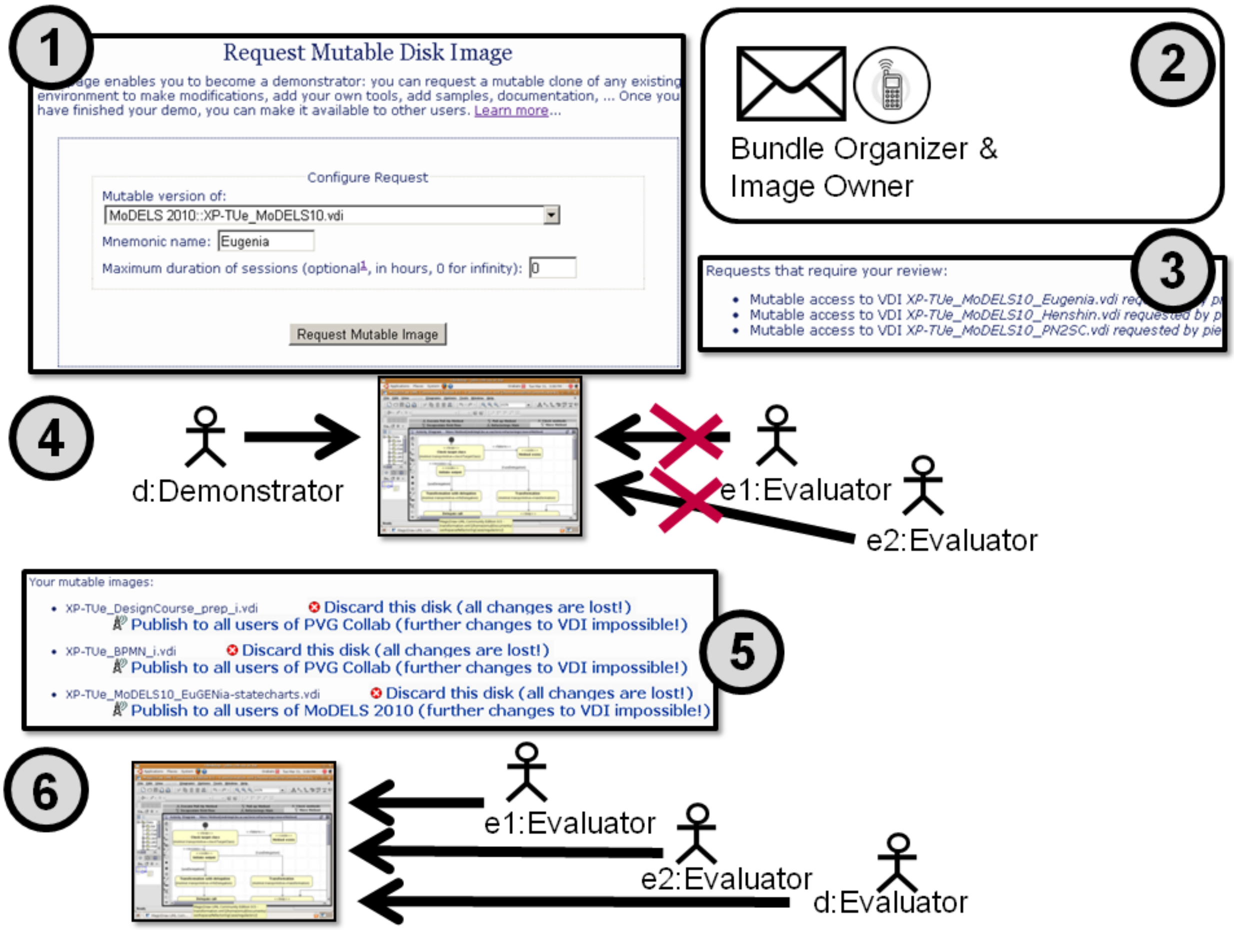
Connection settings

Save current settings, or open saved connection.

Save As... Open...

Connect | Cancel | Help | Options <<

Figure 1: Typical scenario for the reader of a journal special issue or of a workshop proceedings.



Request Mutable Disk Image

age enables you to become a demonstrator: you can request a mutable clone of any existing environment to make modifications, add your own tools, add samples, documentation, ... Once you have finished your demo, you can make it available to other users. Learn more...

Configure Request

Mutable version of: MoDELS 2010::XP-TUe_MoDELS10.vdi

Mnemonic name: Eugenia

Maximum duration of sessions (optional1, in hours, 0 for infinity): 0

Request Mutable Image

Requests that require your review:

- Mutable access to VDI *XP-TUe_MoDELS10_Eugenia.vdi req*
- Mutable access to VDI *XP-TUe_MoDELS10_Henshin.vdi requested by p*
- Mutable access to VDI *XP-TUe_MoDELS10_PN2SC.vdi requested by pie*

Your mutable images:

- XP-TUe_DesignCourse_prep_i.vdi ⊗ Discard this disk (all changes are lost!) Publish to all users of PVG Collab (further changes to VDI impossible!)
- XP-TUe_BPMN_i.vdi ⊗ Discard this disk (all changes are lost!) Publish to all users of PVG Collab (further changes to VDI impossible!)
- XP-TUe_MoDELS10_EuGENia-statecharts.vdi ⊗ Discard this disk (all changes are lost!) Publish to all users of MoDELS 2010 (further changes to VDI impossible!)

Figure 2: Typical scenario for the author of an executable paper based on SHARE.

and the author can restart the workflow. As shown by bullet 6, we assume here that the author publishes the image. All evaluators (e.g., *e1* and *e2* in the figure) as well as the author him- or herself (*d* in the figure) can now start virtual machines for this image, without changing the shared image or seeing each other's changes. This corresponds to the reader perspective, as discussed in Section 2.

5. Related Work (Innovation over Current Options)

Seminal work related to executable papers has been contributed by Claerbout et al. in the early nineties [2]. They already applied automatic build tools to produce CD-ROM images that contained the research article, the corresponding TeX source, related code and data, scripts to rebuild certain figures from the article automatically, and even a special purpose TeX viewer to trigger these scripts while reading the article. Actually, SHARE makes it possible to create a virtual machine with the full content of these CD-ROMs. All of the Claerbout-based executable papers, thus, can be made permanently reproducible inside SHARE.

A more extensive discussion of how SHARE advances the state-of-the-art in the field of reproducible research platforms is provided in [1]. [1] also surveys more recent work that implements Claerbout's ideas[1]. All in all, the most important advantage of SHARE over many other approaches is its flexibility.

[1] See for example `http://www.reproducibility.org`.

6. Conclusion

Several "executable articles" have already been prepared using SHARE. A representative example that highlights potential features of SHARE articles is presented at a companion website for this abstract:

`http://sites.google.com/site/executablepaper/`

Note that a conventional article corresponding to this example has been published very recently by Elsevier [3]. Consequently, it is possible to directly compare the conventional article with its executable counterpart. The latter indeed turns out to be much more useful. We strongly recommend to investigate this machine in order to get an impression of the power of the SHARE approach.

Fulfillment of the Challenge's Criteria

- Executability: SHARE machines can be used very flexibly, so that, among others, interactive equations, tables and graphs are possible and the particular experiment can be repeated and manipulated.
- Short and long-term compatibility: SHARE's only bottleneck with regards to durability is the hypervisor of its underlying virtualization software. Currently, SHARE is built on top of Oracle's (previously Sun Microsystems's) academic version of VirtualBox[2]. But even in the event of discontinuation of that software, SHARE's layered architecture supports long-term availability.
- Validation by reviewers: Opening the environment is just one click, provided the user is logged into the system already. This simplifies the reviewing and the validation of the data and the code.
- Copyright/licensing: SHARE enables authors already to restrict the time that their contributed virtual machine image is used per session. Readers can upload files (e.g., test data) to remote virtual machines. However, they can never download artifacts to their local computers. So far, SHARE has only be used academically. If the Grand Challenge Sponsor aims to make SHARE virtual machines also available to industrial readers, then a special purpose license needs to be developed and agreements with large software vendors have to be made [1].
- Systems: Images are replicated across virtual machine servers. Changes in the infrastructure generally are hidden from end-users. In the domain of high-performance computing, one can make the hardware available as SHARE virtual machine server(s) and restrict the number of concurrent sessions on such servers.
- Size: The SHARE approach saves reviewers/readers from downloading huge files. Moreover, disk usage can be optimized on the server side: large data sets are typically mounted on special network drives that can be read by multiple virtual machines.
- Provenance: SHARE stores information about virtual machine sessions as well as the clone relations between virtual machine images. Such information can be used to perform impact analysis. As discussed in [1], one could install in SHARE virtual machines existing software for tracking events (keybord and mouse actions) to provide more detailed provenance functionality.
- Project quality: SHARE has been stress-tested by the participants of numerous transformation tool contests. At these events dozens of machines run in parallel. The machines containing the submitted solutions are reviewed before as well as during the contest. New machines are created for all solutions of the workshop's live contest and evaluated afterwards. The feedback of the participants regarding SHARE has been very good so far.
- Scope: The aim of the SHARE project is to provide a mature portal for making papers *executable*. Advanced metadata functionality is out of scope, but integration with specialized existing solutions is not. SHARE already provides integration with *LiquidJournal*[3], a platform that enables authors as well as volume editors to combine all artifacts related to a research paper into one integrated online publication that can be analyzed for citations etc.

[2] See `http://www.virtualbox.org/`.

[3] See `http://project.liquidpub.org/research-areas/liquid-journal`.

- Feasibility of integration in publishing workflow and scalability: SHARE's distribution of administrative tasks across multiple organizers is key to the scalability from a publisher's perspective.

7. References

[1] P. Van Gorp, P. Grefen, Supporting the internet-based evaluation of research software with cloud infrastructure, Software and Systems Modeling (2010) 1–18`doi:10.1007/s10270-010-0163-y`.

[2] J. Claerbout, Electronic documents give reproducible research a new meaning, in: Proc. Ann. Int. Mtg Soc. of Expl. Geophys., 1992, pp. 601–604.

[3] S. Mazanek, M. Hanus, Constructing a bidirectional transformation between BPMN and BPEL with a functional logic programming language, Journal of Visual Languages & Computing In Press, Accepted Manuscript (2010) –. `doi:10.1016/j.jvlc.2010.11.005`.