Grant Agreement: 644047

Integrated Tool chain for model-based design of CPSs

HUBCAP

Collaboration Platform v2

Technical Note Number: D5.3

Version: 1.0

Type: Other

Date: June 2021

Public Document

Editors:

Rosamaria Maniaci (ENGIT)

Pietro Greco (ENGIT)

Giuseppe Veneziano (ENGIT)

Lorenzo Sutton (ENGIT)

Reviewers:

Hugo Daniel Macedo (AU)

Gerhard Benedikt Weiß (VV)

Román Felipe Bastidas Santacruz (POLIMI)

Consortium:

Aarhus University	AU	Newcastle University	UNEW
Fortiss GmbH	FOR	Virtual Vehicle Research Center	VV
Fundazione Bruno Kessler	FBK	KTH Royal Institute of Technology	KTH
University “Lucian Blaga” of Sibiu	ULBS	Engineering Ingegneria Informatica S.p.A.	ENGIT
Research Institutes of Sweden AB	RISE	F6S Network Limited	F6S
Politecnico di Milano	POLIMI	Unparallel Innovation	UNP
Controllab Products	CLP	BEIA Consult	BEIA
Verified Systems International GmbH	VSI	Validas	VAL
Technology Transfer Systems srl	TTS		

Document History

Ver.	Date	Author	Description
0.1	15-05-2021	Rosamaria Maniaci, Pietro Greco, Giuseppe Veneziano, Lorenzo Sutton [ENG]	ToC and initial material
0.2	31-05-2021	Ibid.	Initial version of the accompanying report for internal peer review
0.3	31-05-2021	Ibid.	Additional content added
0.4	01-06-2021	Rosamaria Maniaci, Pietro Greco, Giuseppe Veneziano, Lorenzo Sutton [ENG]	Some changes made according to feedback received from peer reviewers. Others to take into account the latest developments
0.9	28-06-2021	Ibid.	Integrating review comments and updates related to release
1.0	30-06-2021	ENG, AU	Final version submitted

Abstract

This report is the accompanying document for deliverable *D5.3 HUBCAP Collaboration Platform Version 2*. The deliverable itself is the **HUBCAP platform release for Month 18** (therefore of type 'Other') and deployed as an online tool available at https://hubcap-portal.eng.it/. Therefore, this document aims to provide an **overview of the results** and artefacts making the deliverable and highlighting the **main features**, with a focus on the **updates since the previous version** (D5.2, version 1).

In particular, the report provides the description of the key **platform features**, also detailing the baseline **needs they respond to**, principles and design choices behind this version of the Platform, along with a description of the **services offered** and the actual ways users can utilise them. In this version a series of **user requirements** coming partners and other work packages (namely WP3 and WP6), have been incorporated and are also highlighted in the report.

List of Definitions and Acronyms

HSM	HUBCAP Sandboxing Middleware
IPR	Intellectual Property Rights
OS	Operating System
DIH	Digital Innovation Hub
SIM	System Interactive Mode
UIS	User Interactive Session
SBM	System Batch Mode
OSP	One Session Passphrase
KPI	Key Performance Indicator

Table of Contents

Table of Figures

List of Tables

1 Introduction

This accompanying report to **D5.3** ***HUBCAP Collaboration Platform v2*****,** released at Month 18 of the HUBCAP project, reports the key updates carried out since the v1 release, highlighting the **development** carried out to incorporate a set of **user requirements** deriving from input and feedback provided by the HUBCAP partners in various ways. The developments between the first and the second release of the HUBCAP Collaboration platform did not require a refinement of the architecture design. The introduction of a new structure for the Collaboration portal workspace, new HSM advantages and new forms of inter-connection between the Collaboration Portal platform and the Sandbox environment are documented in the following sections.

The updated version of the **HSM User Manual** is also provided as an annex (the latest version always being available as an independent document as well).

2 HUBCAP Collaboration Platform: Key updates and features delivered in version 2

2.1 User Journey and updated Workspace Structure and Features

In the second version of the Platform some updates to the Workspace structure have been carried out as explained below.

2.1.1 User journey mapping

Starting from the first release of the HUBCAP Collaboration Platform, the customization work of the DIHIWARE was continued with the design of the HUBCAP workspace story map with the aim to build a shared vision of the digital workspace within their beneficiaries.

Bringing the platform stakeholders (SMEs and DIHs) together around user needs and not around the platform, the design of a high-level user journey has been started. That allowed to understand and determine how users experience the platform based on their unique motivations and goals.

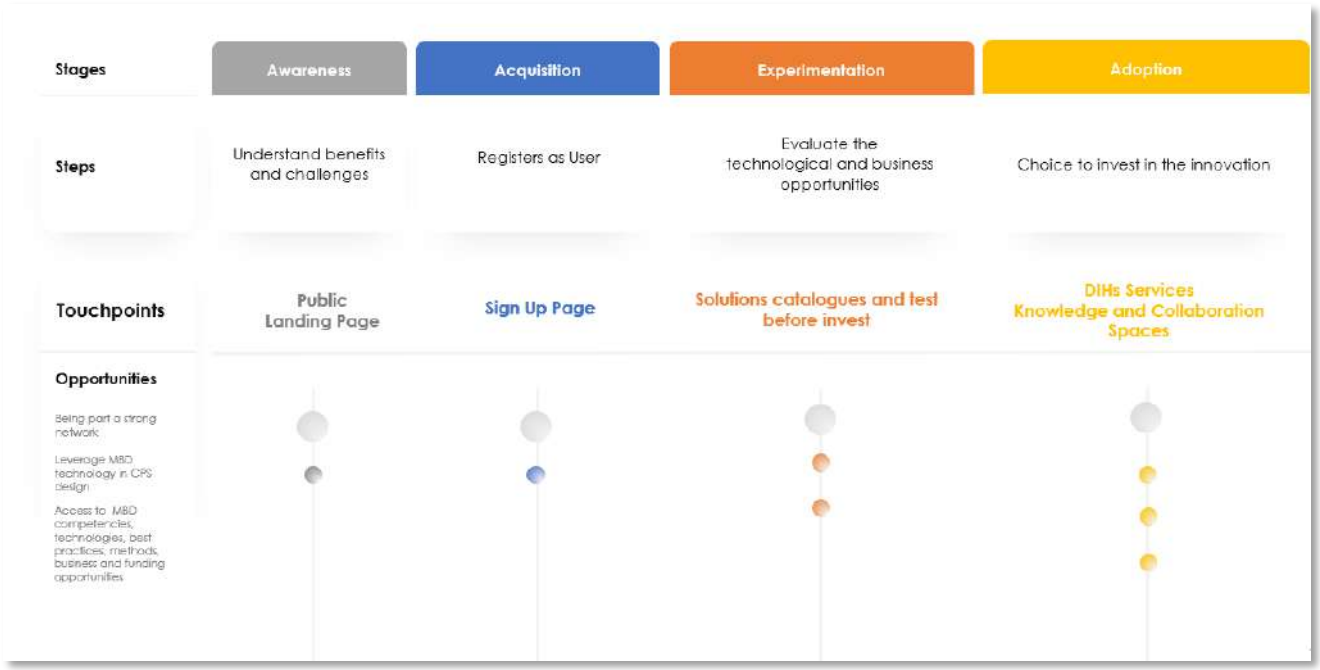


Figure 2-1 User Journey

User journey mapping (Figure 2-1) helps keep user motivation top of mind and create UX flows that get users where they want to go.

With user journey mapping, it is possible to make simple tweaks that help users accomplish their goals easier and faster, come back to do it again, and build habits around the platform.

The first two stages are strictly related to the necessity, raised by the consortium, to engage into the HUBCAP platform external stakeholders (DIHs, Technology providers and Consumers). Those need to understand the full potential that can be achieved by joining the HUBCAP Ecosystem and by active participation in the platform (Awareness) and so register as User (Acquisition).

This is why work has been undertaken on the two main platform touching points (Public Landing Page and Sign-Up Page) that will be available in the next release of the platform and that will enable the first two steps of our User Journey.

Figure 2-2 HUBCAP Platfrom Public Landing Page - First mockup

For the last two stages of the user journey (Experimentation and Adoption) which are already being experienced by our current users (OCs winners), work has been done on the private

section of the platform in order to provide insight and actionable points to begin or improve the user platform experience.

2.1.2 New Private Workspace Structure

During the last couple of months, the team has been working on improving the user story mapping experience with a workspace dashboard and some new spaces.

When users log-in to a workspace, they land on its dashboard. Because there is only one workspace, this dashboard is where they will always land. This intuitive view makes navigation around the story maps easy, given the fact that it gives users an overview on the most important sections within the platform spaces.

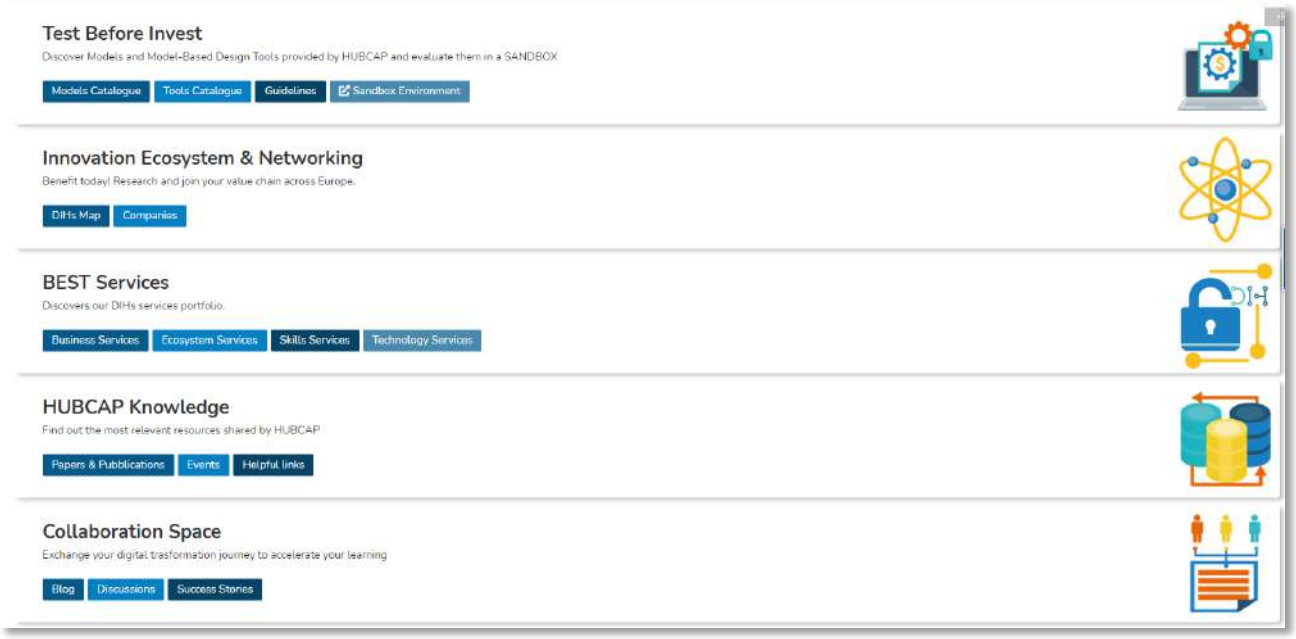


Figure 2-3 HUBCAP Workspace dashboard

As shown in Figure 2-3, the HUBCAP workspace is made by five different spaces that enable our users to go through the different stages which will allow them to meet their needs.

Test Before Invest: The sections (Models Catalogue, Tools Catalogue, Guidelines and Sandbox Environment) empower the discovering of Models and Model-Based Design Tools provided by the HUBCAP ecosystem and their evaluation in the SANDBOX environment. This space enables the provision of *TEST Before Invest* services allowing the experimentation with new technologies – models and tools – to understand new opportunities and return on investments.

Innovation Ecosystem and Networking: The section presents the HUBCAP growing ecosystem made by SMEs, which have market-ready solutions and DIHs offering Services.

BEST Services: The section contains the HUBCAP DIHs Network services offer broken down by type. The Services categorization into the platform is in line with the BEST (Business, Ecosystem, Skills and Technology) model, which is part of the METHODIH made by POLIMI and espoused by

the HUBCAP DIHs network (WP2), containing a set of methods and frameworks (Services portfolio, Customer Pipeline and Business Models) defined to create common standards among DIHs, to support them in service exchange and provisioning.

HUBCAP Knowledge: The section collects the most relevant resources shared by HUBCAP Ecosystem (Papers & Publications, Events and Helpful links)

Collaboration Space: the section allows the exchange of information among the Ecosystem and the possibility to connect with other stakeholders within the platform's ecosystem to significantly increase their opportunities in industry 4.0 journey.

2.2 Tools IPR Protection

2.2.1 Tool Private or Public Visibility

As highlighted by the HUBCAP Project Coordinator, in some scenarios providers might want to make their **tools available only to a restricted set of collaborators** or after having completed some on-going work on the tools themselves.

To meet this requirement, the current version of the HUBCAP Sandboxing Middleware (hereinafter also named **HSM**) gives the tool provider the possibility of specifying the **tool visibility** to either **private** or **public**.

Public tools are visible in the Tools Repository to all the users of the HSM, whereas those marked as **private** are **visible only to their owner** (the tool provider who created them).

The tool owners can specify the tool visibility either during its saving (Figure 2-4) or later from the Tools Repository user interface using the toggle button (as shown in Figure 2-5).

Figure 2-4 Save as a tool section of the control panel. At the bottom, the toggle button for settings tool visibility is shown

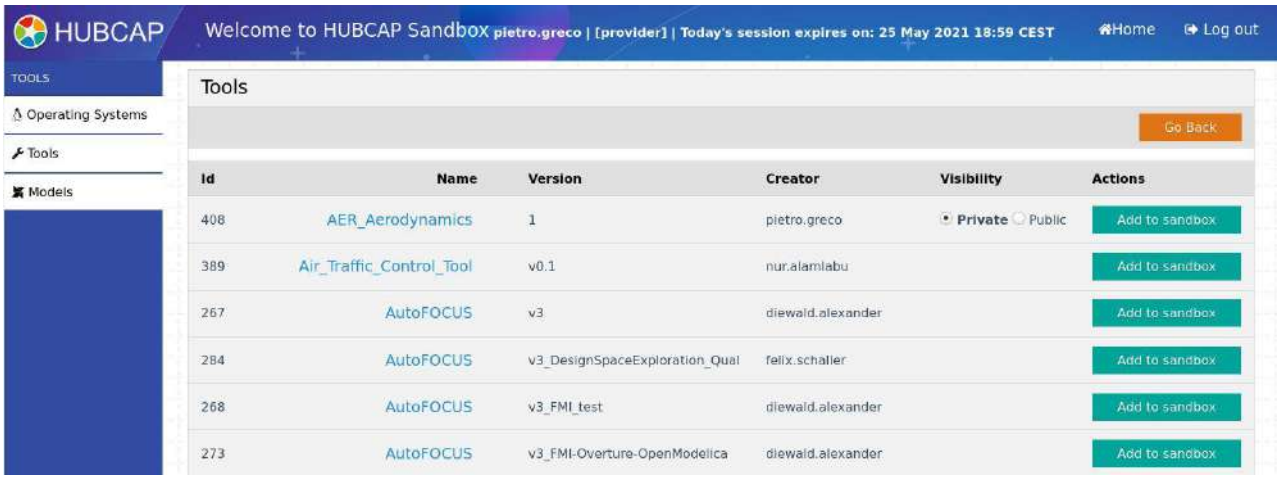


Figure 2-5 The toggle button for settings tools visibility is also shown in the Tools repository

NB: the toggle button appears only for the tools which the provider is owner of.

An interesting consequence of this feature is that the **owner** of a **private tool** can instantiate it on a Sandbox and invite **other HSM users** (acting as guests) **to collaborate inside the same Sandbox.** In such a way, the guests selected by the private tool owner will be able to use a tool they could not otherwise have instantiated on their own (because not visible to them in the Tools Repository). When the **multi-user Sandbox is destroyed**, all the guests will be automatically disconnected and will no longer be able to use the private tool.

2.2.2 Saving new versions of a tool restricted to owner only

Because of IPR policies and tool licenses, **tool owners-providers** want to prevent **other providers from saving customized versions of their tools.** For this purpose, the current release of the HSM ensures that tools – once instantiated within a Sandbox and customized - can only be **resaved by their corresponding owners-providers** while remaining only **instantiable by everyone** else.

2.3 Sustainability

Some aspects (e.g.: price policy) related to the HUBCAP Project sustainability could take advantage of a few HSM technical functionalities. In the current release we have rearranged them and clarified their semantics so that it was possible to make a correlation among HSM users, their profiles and their system resources usage.

In particular, we worked on:

2.a HSM Operational Modes

2.b User Profiles Definition Refinement

2.c Sandbox Low-level Resources Usage Statistics

2.3.1 HSM Operational Modes

Two **HSM operational modes** alternate during each 24 hours: a **System Interactive Mode** and a **System Batch Mode**.

2.a.1. System Interactive Mode (SIM) is a predetermined time slot during which authorized users can initiate a **User Interactive Session** (**UIS**) in order to access – via a WEB GUI – the HSM resources and work on them directly and interactively to manage sandboxes, tools and models.

A new User Interactive Session **starts** when a user logs in to the HSM successfully.

A User Interactive Session **terminates** when one of the following events occurs:

2.a.1.1. - the user **logs out** from the HSM,

2.a.1.2. - the **current SIM** terminates,

2.a.1.3. - the **user remains idle** on the HSM web interface for more than about 15 minutes,

2.a.1.4. - it has been about 10 minutes since an **HSM tab browser closure,**

2.a.1.5. - it has been about 10 minutes since a **browser closure.**

The user **inactivity detection** mechanism which detects these last three events also contributes to **increasing the overall level of security**, as any unattended user session is invalidated.

In order to log in to HSM – thus starting a UIS – a user needs of a **One Session Passphrase** (**OSP**) that is sent to their registered e-mail address. An OSP is a random, **48-char long** string **expiring** when the **UIS terminates** (see the previous five-case list).

2.a.2. System Batch Mode (SBM) is a predetermined time slot during which a user cannot access the HSM WEB GUI. During this time slot the HSM is not sleeping. In fact, some authorized sandboxes - instantiated during a SIM – can survive after the end of the SIM and continue any unattended processing.

*Remark: the **SIM** and **SBM** time slots are **HSM-specific** and are independent from the HSM user profiles and roles.*

2.3.1.1 User Profiles Definition Refinement

In the current release the HSM **users' profiles** definition has been enriched. In fact, alongside the specific **functionalities granted** to each profile (e.g., accessing the OSes Repository, saving a tool instance as a new tool catalogue item, etc.) we have added a new important parameter representing the number of **consecutive SIMs** across which a **sandbox** instance **is preserved** (see table below).

This parameter allows to tune **how long** a user's sandbox can exploit the **low-level resources provided by the HSM.**

In **Annex 1 - *SIM and Sandbox Instance lifetime*** we describe more in depth this topic with some examples.

The following table recaps the current situation for HUBCAP Profile and Role.

Feature	Owner Role		Guest Role
	Provider Profile	Consumer Profile	Whatever Profile
Consecutive SIMs	3	1	-
Access to remote viewer	X	X	X
Upload Archive	X	X	-
Download Archive	X	X	-
Invite Guests	X	X	-
Destroy Sandbox	X	X	-
Select Tool	X	X	-
Select Model	X	X	-
Select Operating System	X	-	-
Save a New Tool	X	-	-
Save New Tool Version	X (their own)	-	-
Upload New Model	X	-	-
Delete Repository Item	X (their own)	-	-

Table 2-1 Feature table illustrating the functionalities available based on the Profile/Role combination

2.3.1.2 Low-level Resources Consumption Statistics and Monitoring

The current HSM release greatly improves the collection of data regarding **low-level resources consumption**. Detailed and granular statistics about this aspect are reported for **each asset running** inside a **sandbox instance** (tools and shared storage).

The **per-asset** monitored parameters, extracted directly by the operating system hosting the tools virtual machines, are listed below and are **stored in the HSM logging system** when a sandbox is destroyed:

- **CPU_sec**: CPU cumulative usage time in seconds;
- **RAM_gb**: allocated RAM in GBytes;
- **Cores_#**: allocated cores number;
- **NET_IO_kb**: globally exchanged network traffic in KBytes;
- **DSK_RW_mb**: globally disk read/written data in MBytes;

Moreover, the **sandbox owner** can take a look at them while the sandbox **is running** as well as when **it gets destroyed**.

In the first case the values will not be the definitive ones (Figure 2-6). In the second case the values will be as those stored in the HSM logging system (Figure 2-7).

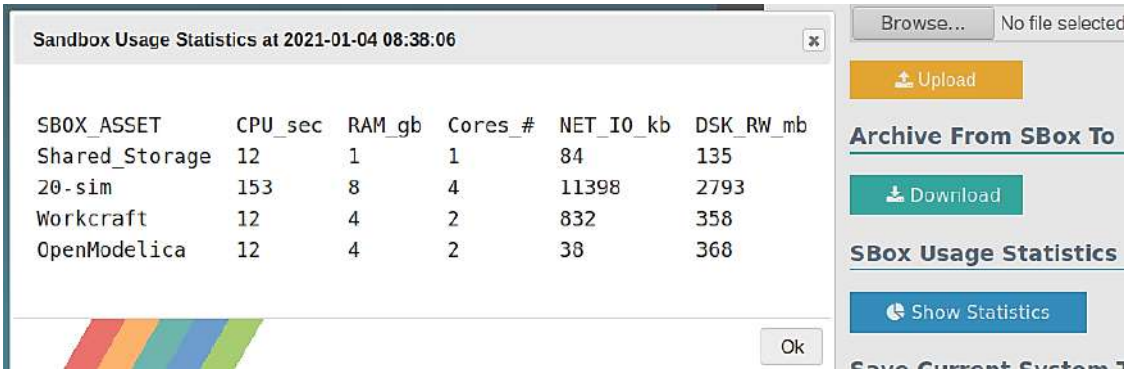


Figure 2-6 Dialog showing current resources consumption

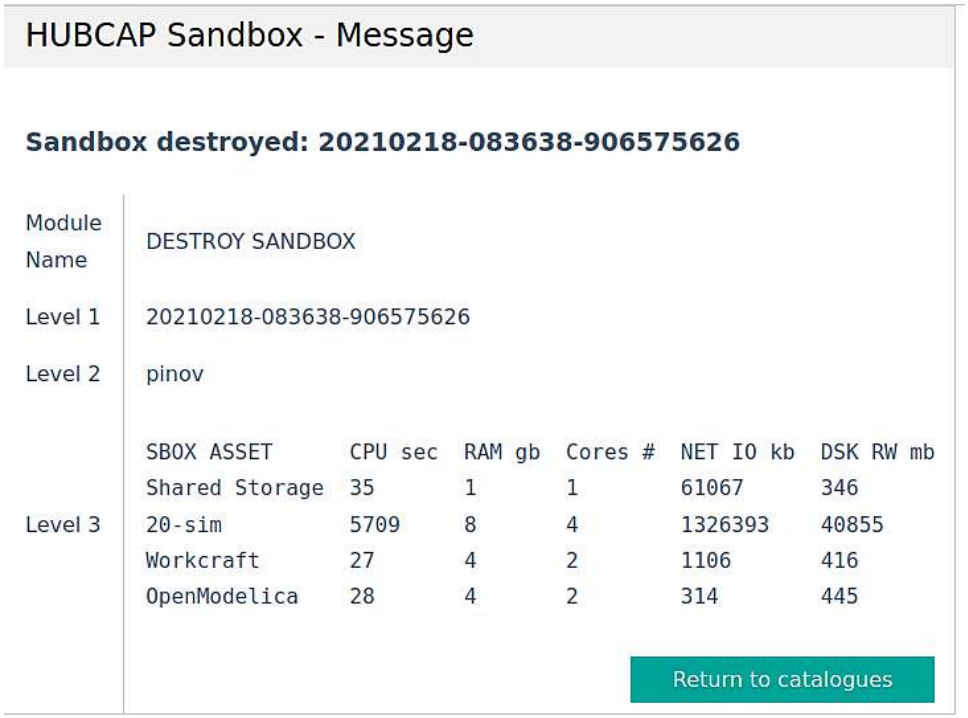


Figure 2-7 Consumed resources shown upon Sandbox destruction in the action confirmation message

Obviously, the collection of these sandbox-related values can be aggregated to obtain a few **monitoring functionalities** currently used at system administrator level specially to detect the entire HSM's load.

2.3.1.3 Support for initial pricing strategies

Initial discussions among partners started about future pricing strategies and models aimed at longer-term (i.e., after the project end), sustainability of HUBCAP. The features described up to

now can also support some of the suggested pricing policies, which could cover – at least - the **infrastructure costs** (e.g., the monthly rent of the needed physical servers).

Example 1

The new profile parameter (**number of consecutive SIMs**) takes into account the **greater or lesser possibility of using the server physical resources via HSM**. It gives the chance to define more profiles, each one with a **fixed fee** based on it.
Remark - Currently, there are two HSM profiles: Provider with 3 consecutive SIMs and Consumer with 1 SIM only.

Example 2

The new **per-asset resources consumption statistics** allows to know how much a user loads the system both overall and per–tool. Based on that, a **variable share** could be calculated taking into account the **really used** low-level resources.

Example 3

The HSM usage price could be a combination of the **fixed** and **variable share.**

A more detailed example

In current KPIs calculations (see next section), among the others, two sorts of outcomes (**Users' Billing** and **Tool Profitability**) are provided by aggregating these parameters on a per-user and per-tool basis and monetizing them with the following unitary values:

Parameter	€/Unit
CPU_sec	**0.00100**
NET_IO_kb	**0.00001**
DSK_RW_mb	**0.00100**

Table 2-2 Tentative estimation of low-level resources unitary prices for a PoC implementation

Obviously, these values are there only for a mere PoC and not for an application in the real world.

Users' Billing

This is just an example to demonstrate how a price can be built based on how many system resources a user has consumed.

Resources Usage per User - Year 2021	CPU time [secs]	Network I/O [KB]	Storage R/W [MB]	Bill [€]
alexandru.vulpe	755	408700	2038	6.88
amos.smith	2260	1291385	8387	23.56
b.gunnink	810	407907	2643	7.53
barner.simon	135	2928	1641	1.81
daniel.craciunean	644	4021117	6043	46.90
dario.pietraroia	153998	1268725	3167	169.85
frank.zeyda	7846	4255285	16463	66.86
frank.zeyda2	306	252638	2064	4.90
george.iordache	3592	1327702	13227	30.10
george.suciu	14648	958314	6194	30.43
gerhard-benedikt.weiss	381128	4882826	27747	457.70
pietro.braghieri	485	65365	4563	5.70
rajivarnan.r	726	412162	2076	6.92
stefano.tonetta	53	487	1067	1.12
TOTAL	626640	39724476	244831	1268.72

Figure 2-8 Resources Usage per User

and what is the contribution to the total revenue.

For a more general price building, a fee depending on the user profile (e.g.: based on the number of SIMs assigned to the user's profile) could be taken into account too.

Tools' Profitability

This is just a simple example to demonstrate how a tool profitability indicator can be built to take into account how much a tool contributes to the total revenue. For a more general profitability index, also the tool efficiency and other more complex parameters should be weighted.

Resources Usage per Asset - Year 2021	CPU time [secs]	Network I/O [KB]	Storage R/W [MB]	Bill [€]
ST - Shared_Storage	22828	28621941	104052	413.10
OS - centos7	91	69475	745	1.53
OS - centos7_XFCE	4399	2063514	8654	33.69
OS - ubuntu18_XFCE	16650	635319	20080	43.08
TO - Air_Traffic_Control_Tool	779	145585	3522	5.76
TO - AutoFOCUS	420	8675	1738	2.24
TO - AutoFOCUS3	4848	3542	1868	6.75
TO - Beia_Tool	1726	24717	2537	4.51
TO - test_browser2	26130	21850	2111	28.46
TO - Test_by_Benedetto	964	23913	2710	3.91
TO - Test4NurAlam	4062	962634	15551	29.24
TO - tool4Nur_SDE	4606	69319	5239	10.54
TO - tool4Nur_SDE_Chess	7473	55925	5190	13.22
TOTAL	626640	39724476	244831	1268.72

Figure 2-9 Resources Usage per Asset

2.4 KPIs Evaluation

Detailed statistics related to the HUBCAP Platform have been calculated and made available to the authenticated users via a dedicated button in the side bar of the Collaboration Portal.

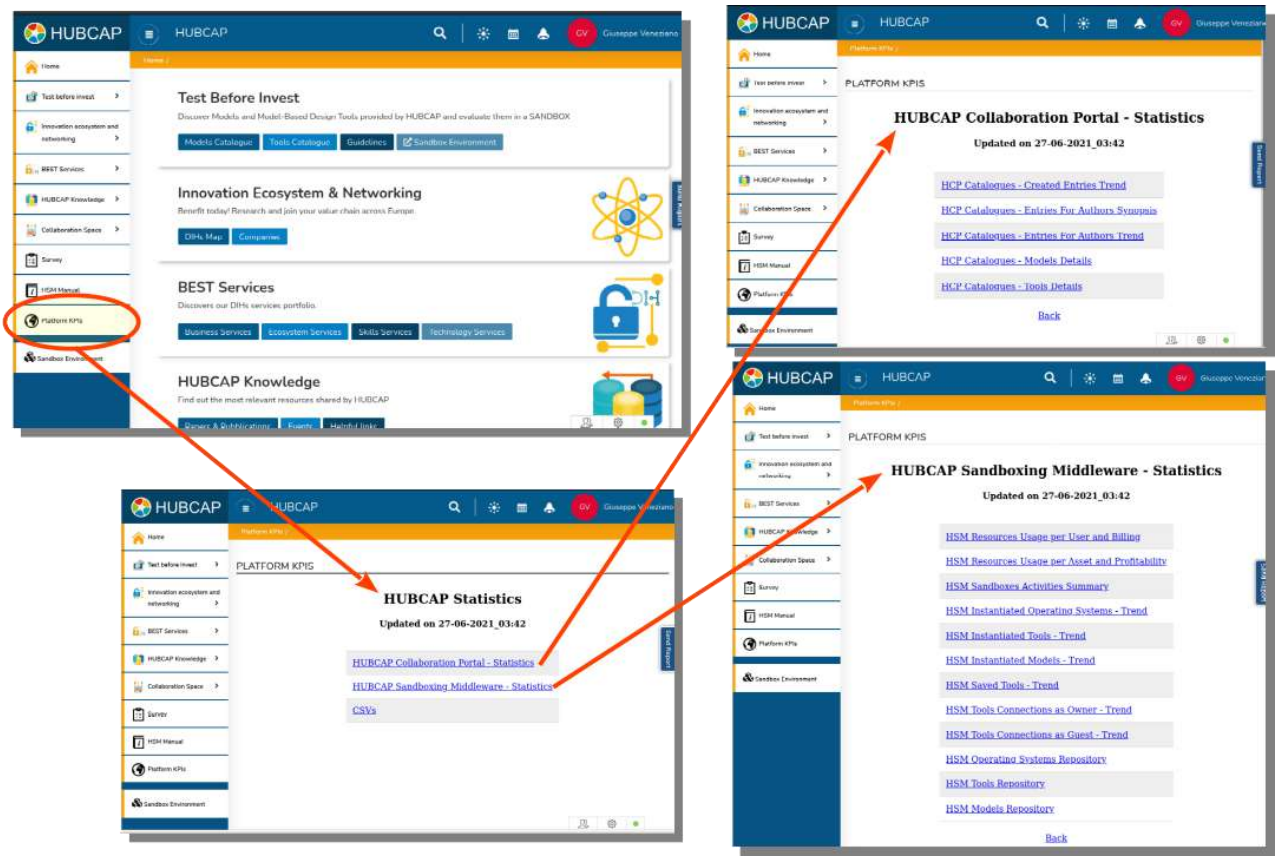


Figure 2-10 All-in-one short illustration about how to access the KPIs and the main navigation paths

The current reports are grouped for the HUBCAP Collaboration Portal (HCP) and for the HUBCAP Sandboxing Middleware (HSM), as follows:

HUBCAP Collaboration Portal

1. Catalog_Created_Entries_Trend
2. Catalog_Models_Details
3. Catalog_Tools_Details
4. Catalogues_Entries_For_Authors_Synopsis
5. Catalogues_Entries_For_Authors_Trend

HUBCAP Sandboxing Middleware

1. Instantiated_Models_Trend
2. Instantiated_Operating_Systems_Trend
3. Instantiated_Tools_Trend
4. Models_Repository

5. Operating_Systems_Repository
6. Resources_Usage_per_Asset_and_Profitability
7. Resources_Usage_per_User_and_Billing
8. Sandboxes_Activities_Summary
9. Saved_Tools_Trend
10. Tools_Connections_as_Guest_Trend
11. Tools_Connections_as_Owner_Trend
12. Tools_Repository

Here are two examples of these reports, one for each environment:

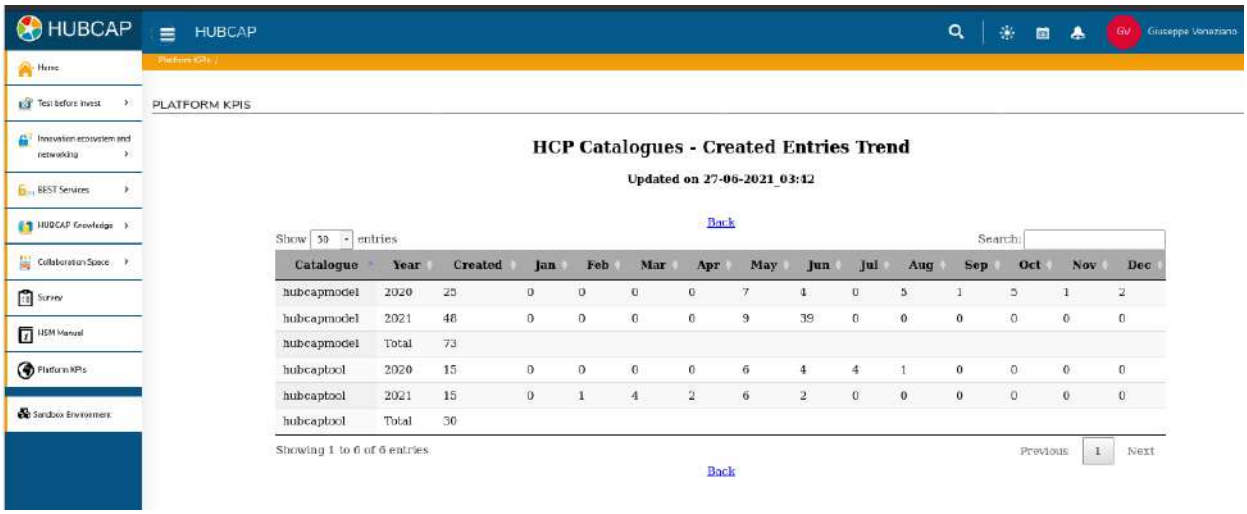


Figure 2-11 Report illustrating when new models and tools have been added to the Collaboration Portal Catalogues

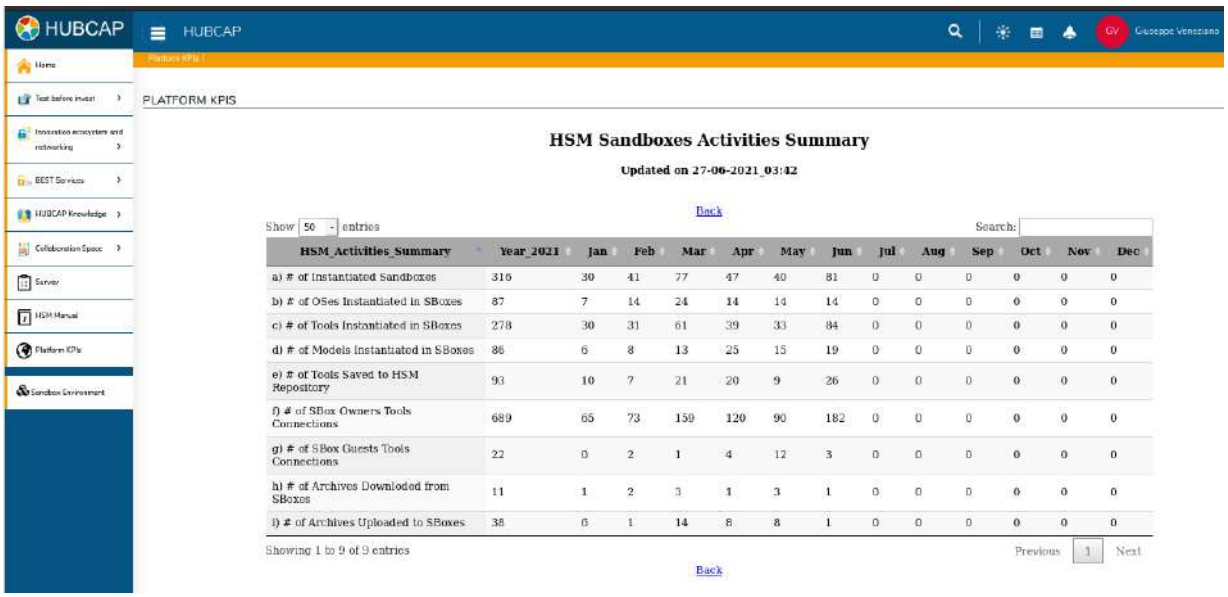


Figure 2-12 Report illustrating the main HSM activities distributed over the months

Moreover, a CSV version of all the reports is downloadable using the "CSVs" entry (from the main KPIs index) bringing to this simple page:

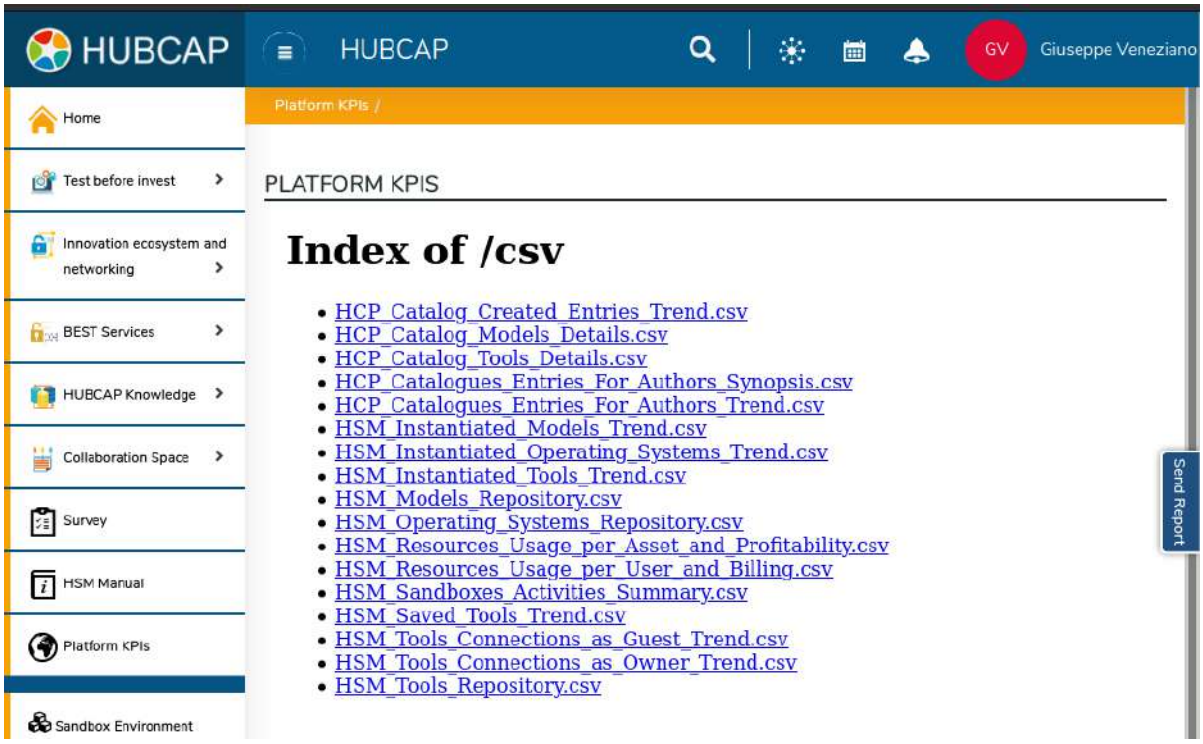


Figure 2-13 List of all CSV files available for download

All these topics are covered more in detail in the **HUBCAP HSM User Manual** (section 7. KPIs).

2.5 Sandbox Users Activity Logging

Besides logging low-level resources consumption, the HSM also tracks user's **activity for each Sandbox Instance**. This information is stored as a set of events in log files.

Below is a list of the **currently tracked events**:

- Start of a Sandbox
- Destruction of a Sandbox
- User (owner or guest) resumes remote desktop connection the Sandbox tools
- E-mail warning about Sandbox expiration is sent to the owner
- User has established a remote desktop connection to a specific tool in the Sandbox
- Sandbox owner enables a guest to access their Sandbox
- Sandbox owner revokes access to their Sandbox to a guest
- User has left the Sandbox Viewer
- Sandbox Owner saves a customized version of a tool or operating system

Finally, a **timestamp** - in the format: ***YYYYMMDD-HHMMSS*** – is associated to each Sandbox Instance event.

2.6 Lowering Barriers to Experimentation

In this section we present features which are particularly aimed at lowering the barriers to experimentation and provide improved user experience and access of functionalities to users within the platform. These features were developed in tight collaboration with all HUBCAP partners and through a series of "feedback" loops where features where suggested, tested and refined.

2.6.1 'Try It Now' functionality

Wanting to enhance the user experience in the use of the Collaboration Platform, a new functionality named "**Try It Now**" has been implemented. In particular, it makes it easier for the end-users to **switch** from the exploration of the **HUBCAP Models and Tools Catalogues** to the actual **use of models and tools in the HUBCAP Sandboxing Middleware**.

2.6.1.1 HUBCAP Collaboration Portal Support for Models

For models, this feature can be enabled by the **providers** by adding the following information to the details of a model inside the Models Catalogue (Figure 2-14):

1. a reference to the **model** uploaded to the HSM,
2. a reference to a **tool** already available in HSM and **able to process the model**,
3. a reference to a step-by-step **user guide** which explains how to use the model with the associated tool inside a same sandbox.

Figure 2-14 Model dialog showing the information needed to enable the Try it Now feature

If **all the three** above mentioned fields have been filled, the system will show the **end-user** viewing the model entry in the HUBCAP Models Catalogue, a "Try it Now" section (Figure 2-15).

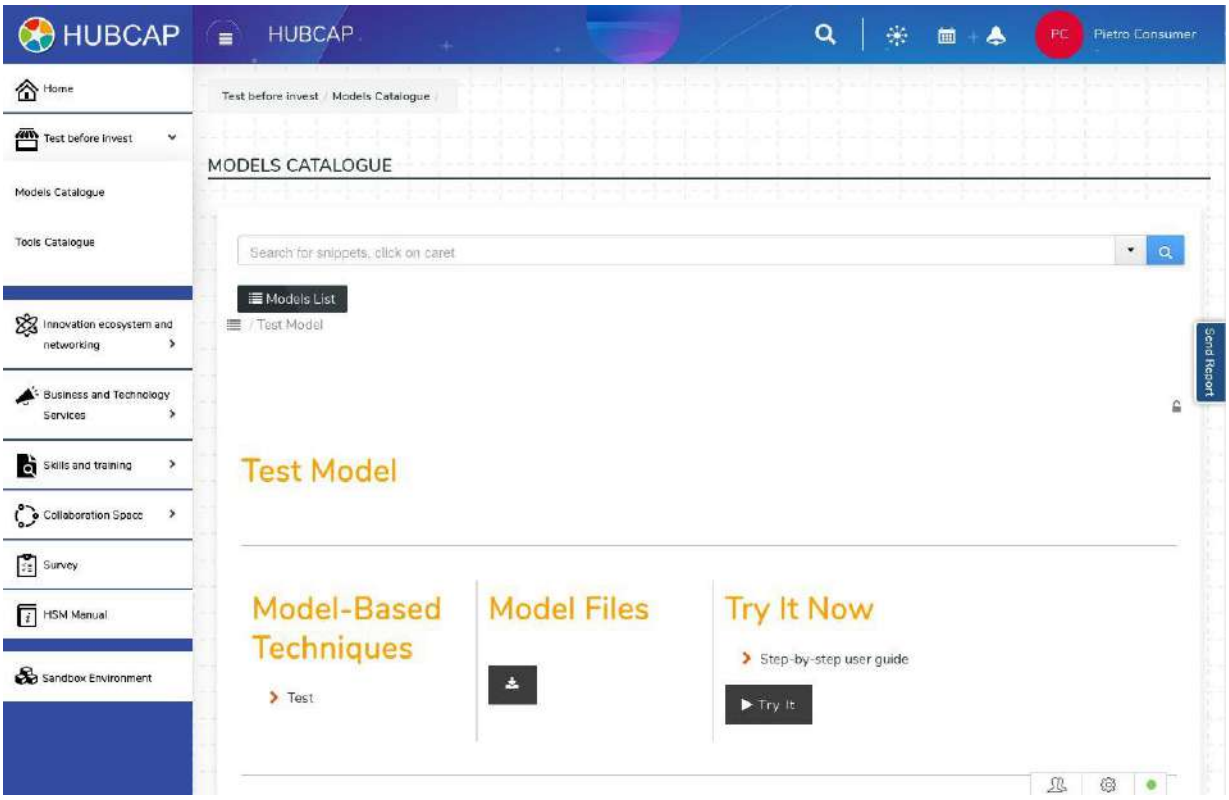


Figure 2-15 Model details page with Try it Now section

Clicking on the "Try It" button, the HUBCAP user (after the usual HSM authentication) will find the sandbox cart automatically filled with both the model selected from the Models Catalogue and the tool pre-set for it by the model provider (Figure 2-16). Clicking on the **"Step-by-step user guide"** link, the user guide will be opened for an easy consultation.

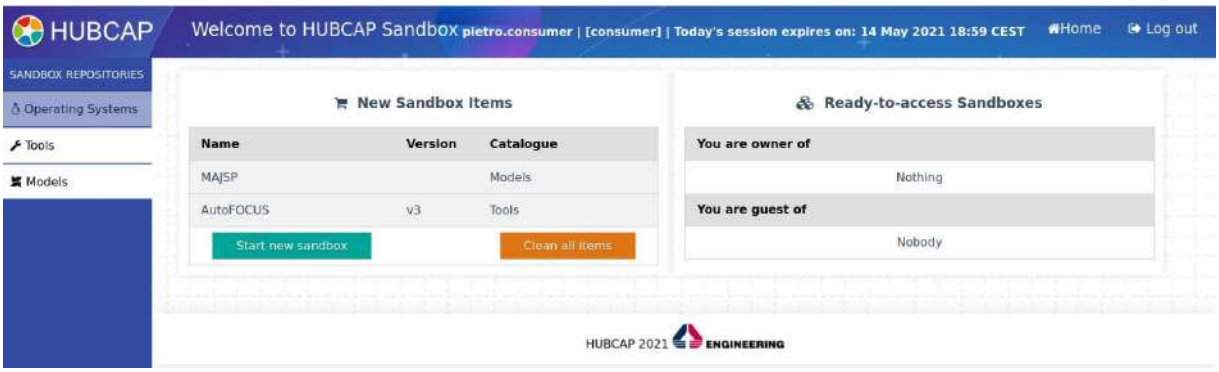


Figure 2-16 HSM homepage showing the sandbox cart ("New Sandbox Items" section) which has been automatically filled through the Try It Now function

2.6.1.2 HUBCAP Collaboration Portal Support for Tools

The same function is also available for Tools. In this case only the Tool ID has to be added to the details of a tool inside the Tools Catalogue to enable the functionality.

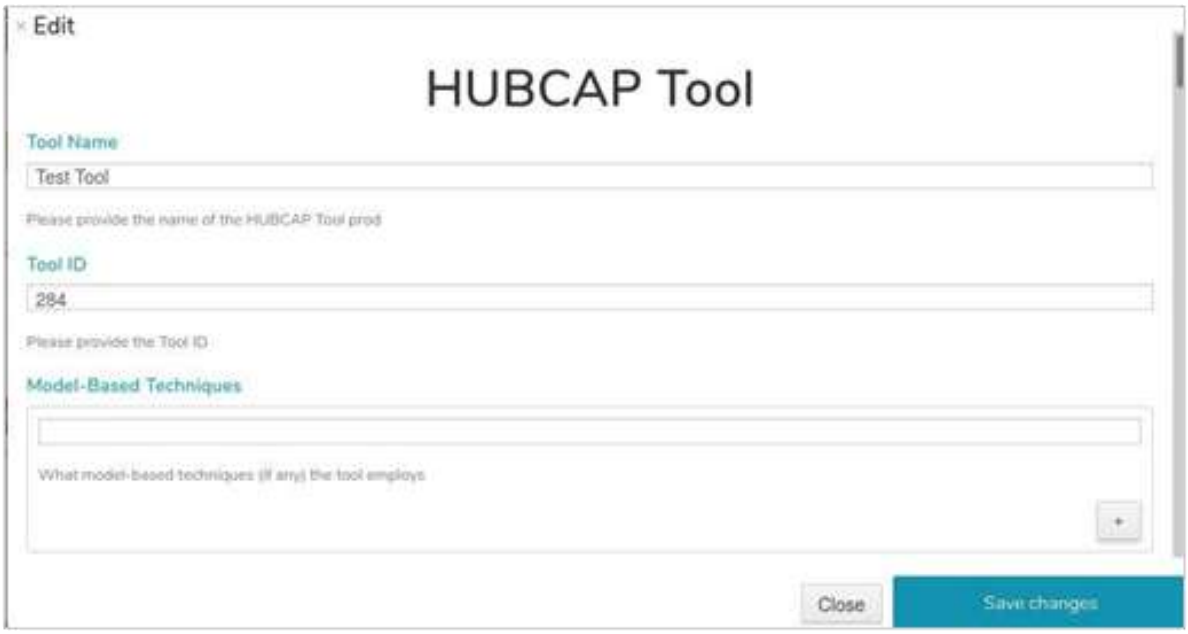


Figure 2-17 Tool Create/Edit Form available in the Tools Catalogue portal side

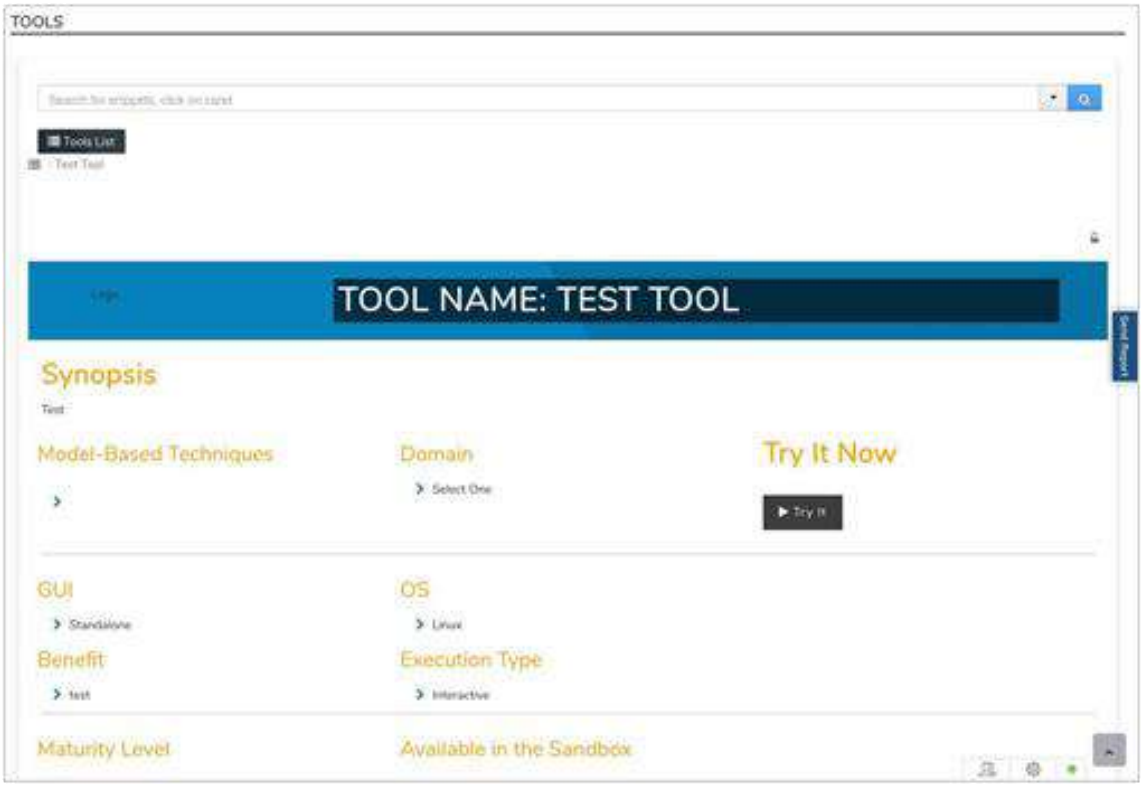


Figure 2-18 Tool Details page with active Try It Now section

In this case, clicking on the "Try It" button, the user will find the sandbox cart automatically filled with only the tool.

2.6.1.3 HUBCAP Sandboxing Middleware Support for Models and Tools

The ***Try It Now*** functionality for models and tools creates a 'shortcut' for users from the HUBCAP Models and Tools Catalogues to directly reach the HSM. Below we are going to briefly describe how the Sandboxing Middleware has been updated to support this functionality.

First of all, **tool IDs** and **model IDs** are now **shown in the HSM Repositories** (see Figure 2-19) so that the model providers can fetch and use them to populate the model details in the HUBCAP Models Catalogue.

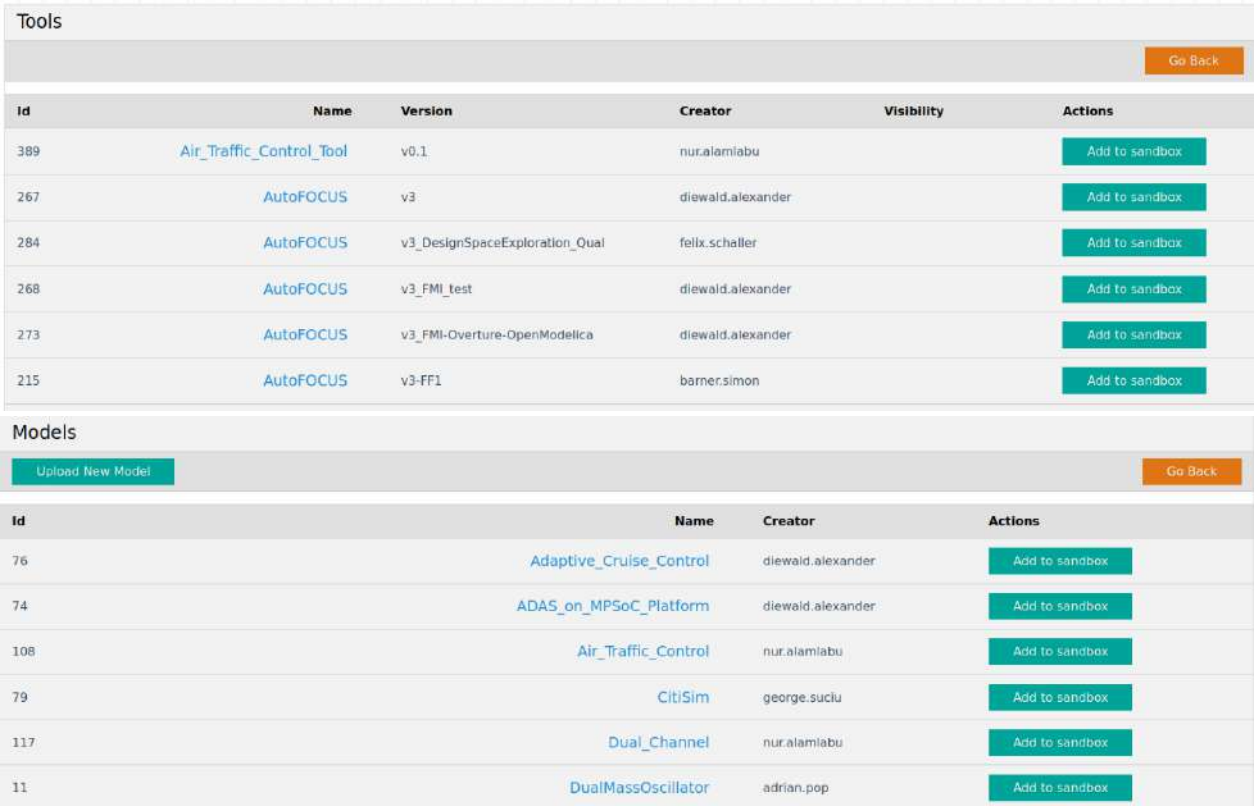


Figure 2-19 Tools and Models Repositories showing on left the tool/model ID

In addition to this, the Sandboxing Middleware also exposes an **additional REST endpoint** through which it receives from the Collaboration Portal the information required to fill the sandbox cart.

The logic of this endpoint holds in great esteem the robustness and reliability of implementation as well as a smooth end-users experience in switching from the Models or Tools Catalogues to the Sandboxing Middleware.

2.6.2 Improvements to the User Interface

2.6.2.1 Dialogs

As part of the work done to **improve the user experience**, the information dialogs appearing after clicking on the "Sandbox Environment" button now include a **structure and messages** revisited to be **clearer** (see *Figure 2-20*).[1] Moreover, the current release of the Portal ensures that **users read the messages before entering the Sandbox Environment** by making them click on a confirmation button.

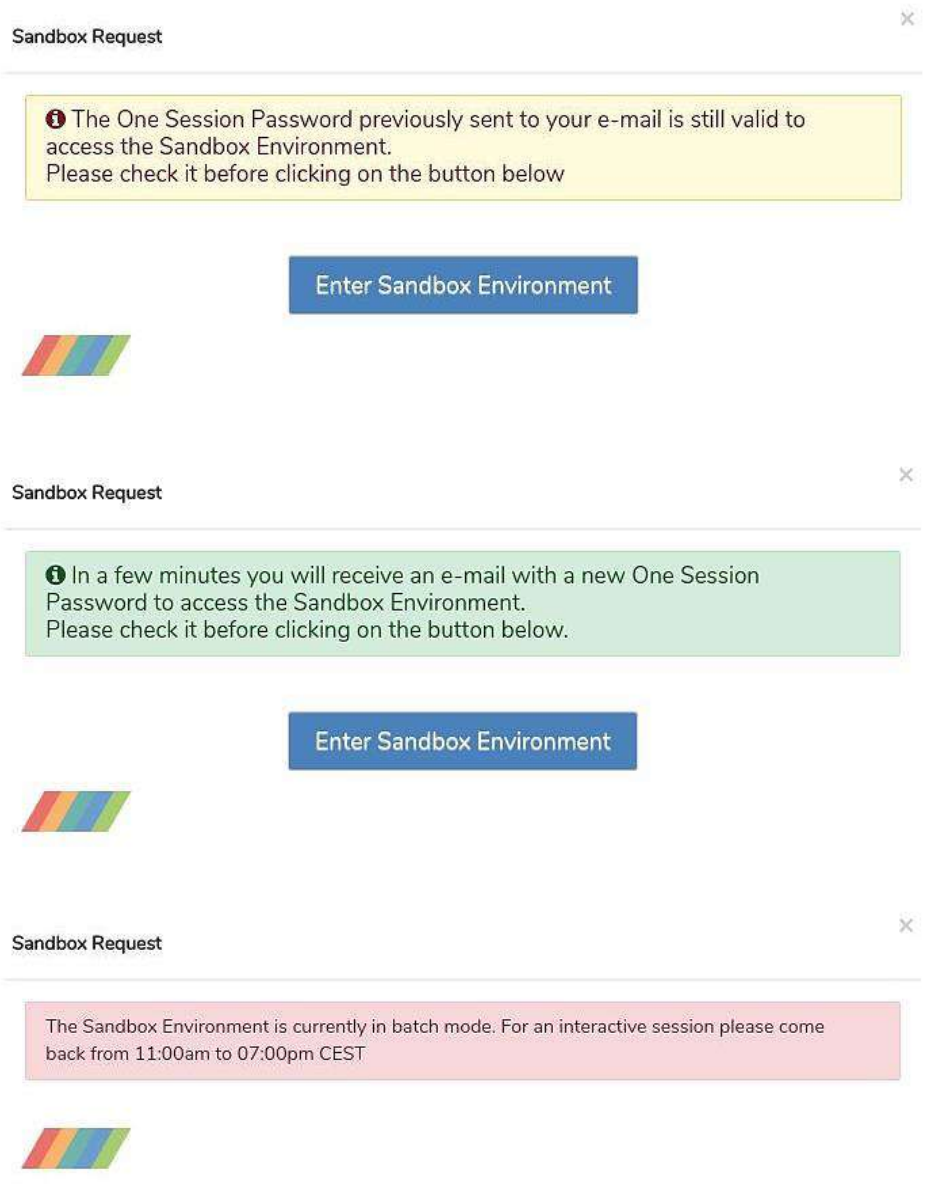


Figure 2-20 *Revisited information dialogs featuring clearer structure and messages*

[1] We would like to particularly thank Frank Zeyda [VSI] for suggestions and feedback about these improvements and features.

2.6.2.2 Sandbox Viewer

The current release of the HSM comes with a set of improvements, optimizations and fixes to minor issues.

To start with, the Sandbox Viewer (Figure 2-21) – the portion of HSM Web User Interface that allows end-users to interact with Sandbox and tools running in it – has undergone some refactoring aimed to increase the **maintainability** of the web page and decrease its **loading times**.

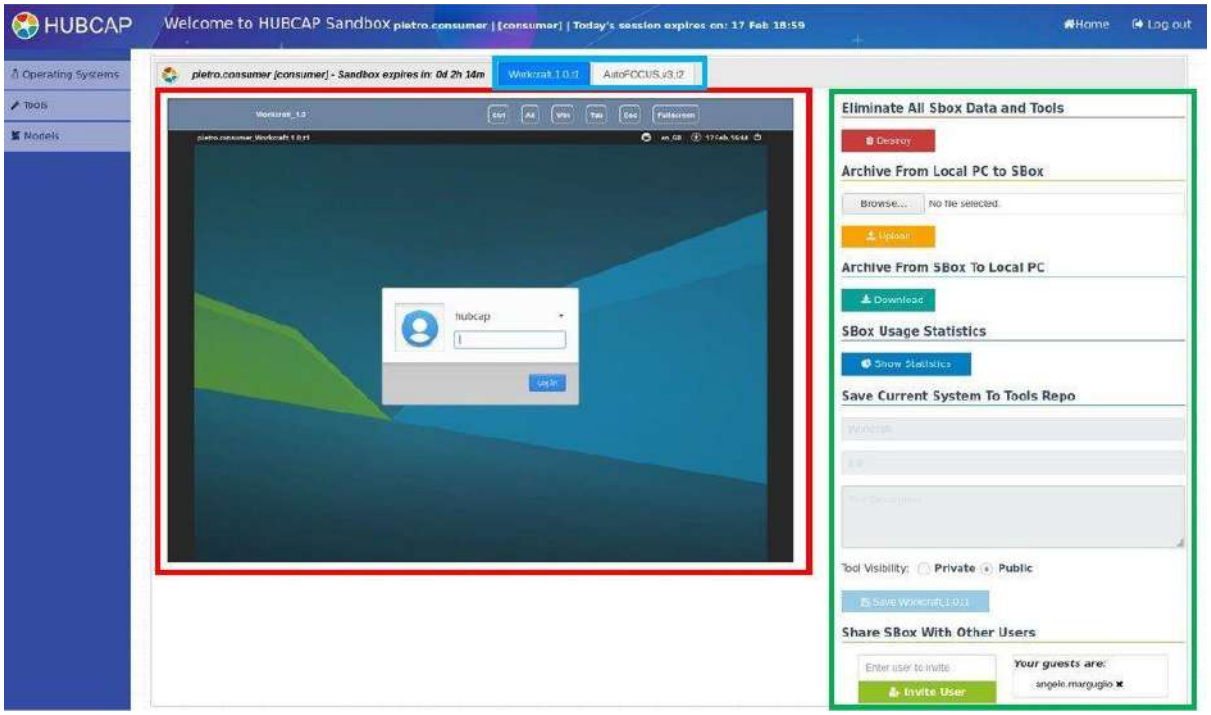


Figure 2-21 Sandbox Viewer

Other changes to the Sandbox Viewer concern the **Sandbox Sharing Panel (***Figure 2-22***)**. The list of candidate Sandbox guests has been replaced with a more user-friendly **autocomplete-enabled search bar** that makes it easier to browse and select guests.

Figure 2-22 *Detail of the new Sandboxing Sharing Panel with autocomplete-enabled search bar on the left and list of enabled guests on the right*

Following the implementation of the low-level resources monitoring module for the HSM (described above in section 2.c), the Sandbox Viewer now also gives the end-users the possibility of **visualizing in real-time a summary of the computational resources consumed by their sandbox,** namely by the Sandbox Shared Storage and by each Tool running in it.

Clicking on the "Show statistics" button available in the control panel (*Figure 2-23*). the user (sandbox owner) visualizes, in a pop-up dialog:

- CPU time
- Allocated RAM
- Network traffic
- Disk Read/Written Mbytes

Figure 2-23 *The Show Statistics button is available in the control panel*

This is also illustrated in the following Figure 2-24

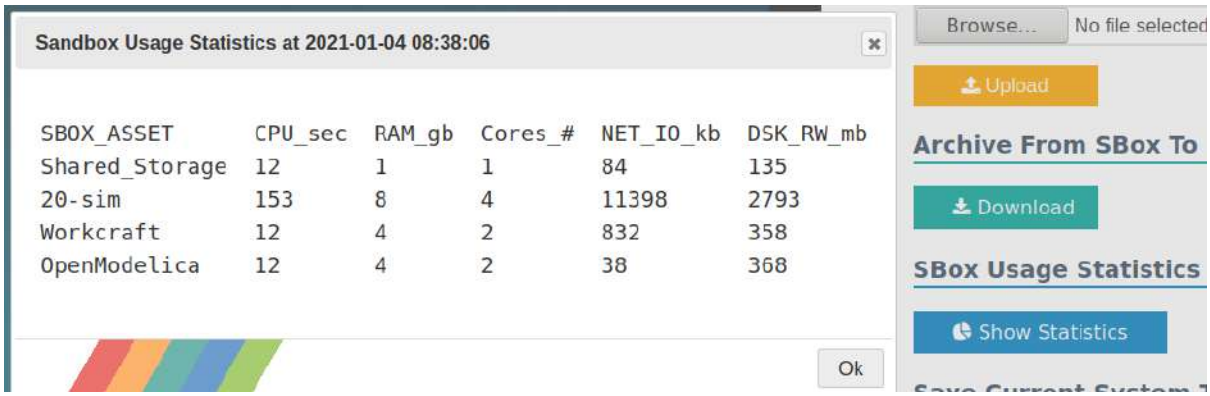


Figure 2-24 Dialog showing the currently used computational resources

The same information is now also shown upon Sandbox destruction in the action confirmation message (*Figure 2-25*).

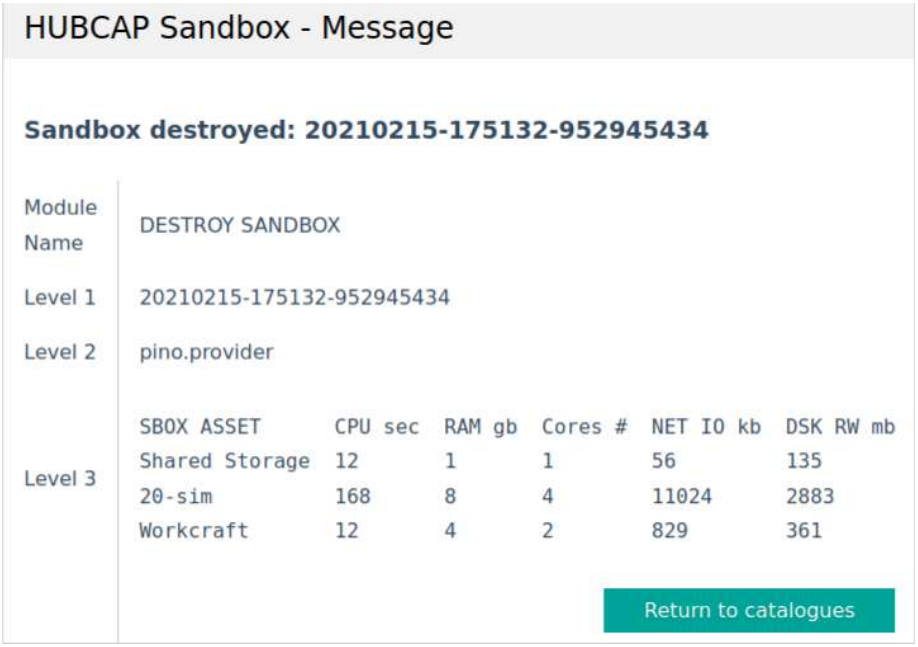


Figure 2-25 *Sandbox destruction confirmation message containing a recap of the consumed resources*

2.6.2.3 Sandbox Viewer - Control Panel

The **control panel** within the Sandbox Viewer **is no longer shown to sandbox guests,** thus avoiding exhibiting a lot of disabled control buttons (Figure 2-26).

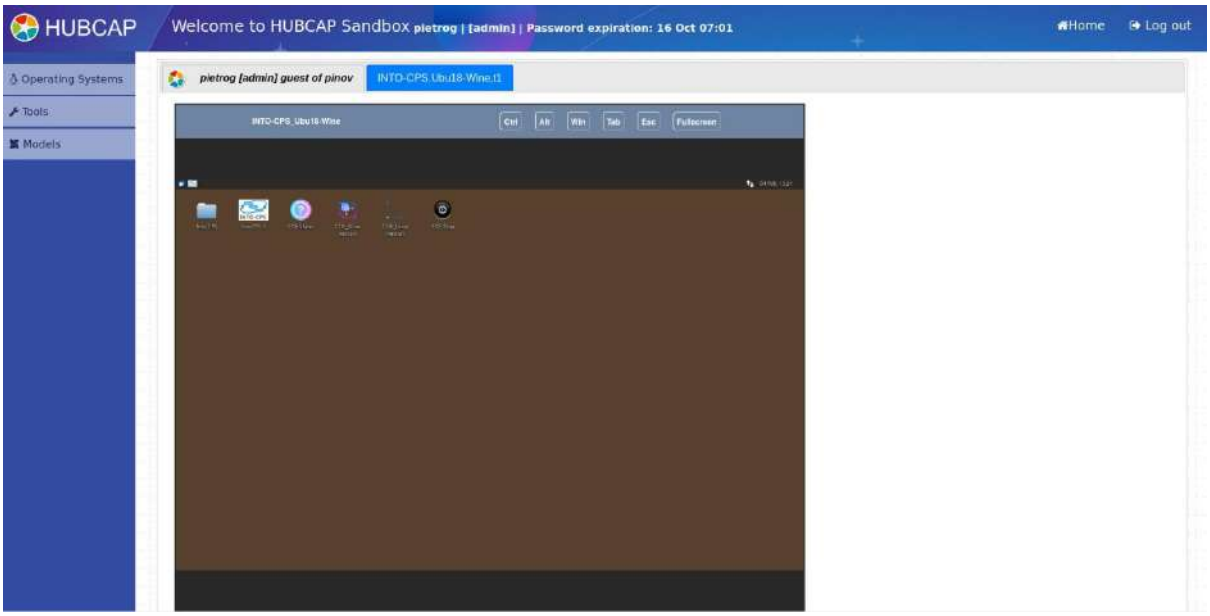


Figure 2-26 Sandbox Viewer as appears to a guest; no control panel is shown on the right

2.6.2.4 Tools Repository Normalization

Other changes to the user interface followed the normalization of the Tools Repository. As it is possible to see in the figure below (Figure 2-27).

Figure 2-27 Tools Repository showing the new columns: ID, Name, Version, Creator and Visibility

Name and version are now shown in two separate columns and a Creator column has also been added.

2.6.2.5 User Inactivity Detection

As mentioned in section 2a, this new release of the middleware also features a user inactivity detection system, which also **increases the level of security** by invalidating unattended user sessions.

User interface side, the HSM now informs the users about the incoming User Interactive Sessions (UIS) expiration through the following pop-up message (*Figure 2-28*).

Figure 2-28 *Dialog shown to the end-users when their User Interactive Session is about to the invalidated*

The system also ensures that users have read the message by only allowing them to **prolong the session** by clicking on either the Close or Ok buttons.

At the expiration of the user interactive session, the HSM also closes the tab thus guaranteeing **increased confidentiality.**

2.6.2.6 E-mail Messages

Also, the e-mail message sent to the users to communicate the **One Session Password** (see section 2.a) has been revised to include **more detailed and punctual information about session expiration** (see *Figure 2-29* below).

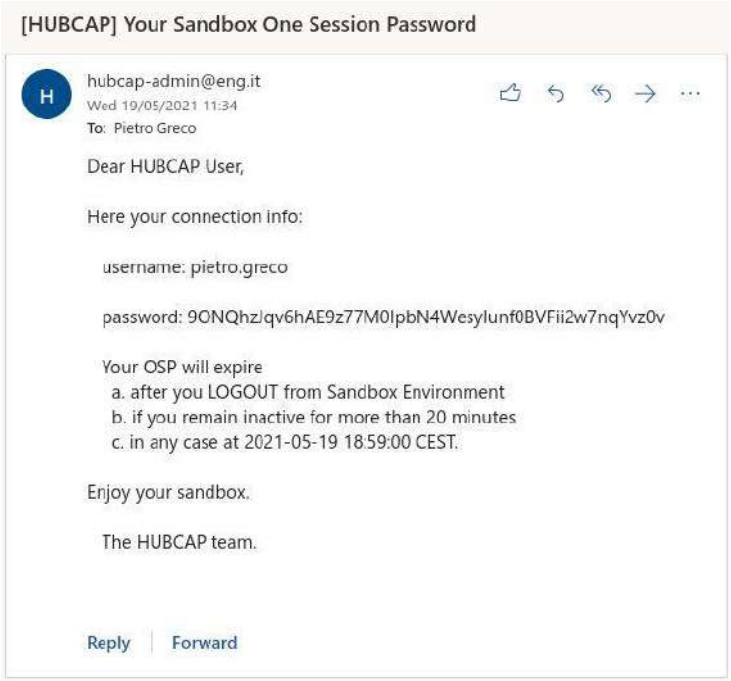


Figure 2-29 *Revised email message sent to the users after they request access to the HSM*

2.6.3 Bug fixes

Actions were also taken to solve some minor issues related to the display of information about session expiration **datetime** and **time zone** (Figure 2-30).

Figure 2-30 Top bar where information about user and session is displayed

2.6.4 User Guides

The HUBCAP Sandboxing Middleware user guide has always been kept up to date and enriched since the initial versions of the HUBCAP Platform, also taking into account feedback received from the partners and requests for material which is useful for users to access and use the tools. In fact, the HSM User Manual (which is provided as Annex at the end of this report), is also accompanied by a series of video tutorials produced by other WP partners which provide intuitive hands-on demos and guides on how to use the Platform.

Annexes

Annex 1 - SIM and Sandbox Instance lifetime

In this annex we explain in more detail how the Sandbox instance lifetime works.

Example 1

A **consumer user** during his UIS creates a SBox1, then destroys it and creates a SBox2 that the consumer will be able to use till it is **destroyed automatically** at the **end of the one and only SIM** assigned to his profile (Diagram 1).

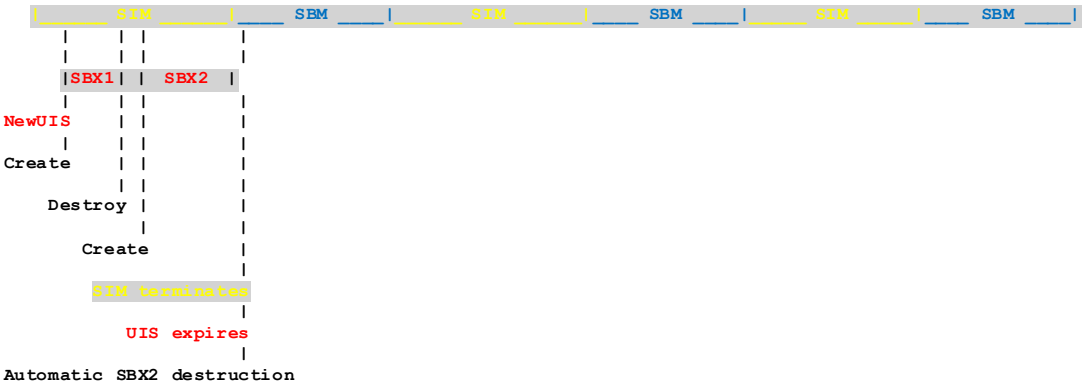
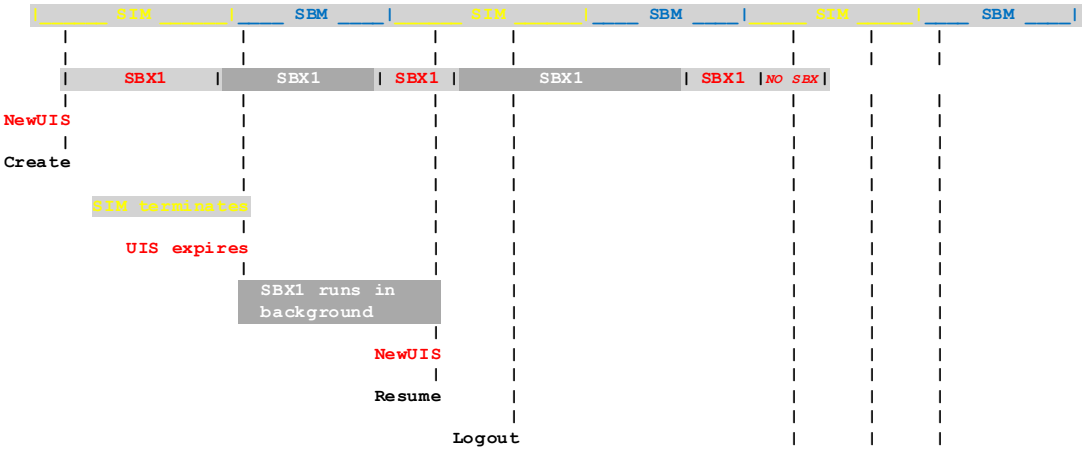


Diagram 1 – An example of how a consumer user may use his or her single SIM

Example 2

A **provider user** starts a SBox1 at his 1st SIM and uses it till the 1st SIM terminates, at which point his UIS expires together with his OSP. However, the SBox1 keeps running also during the 1st SBM and the provider user will resume the interactive session to interact again with it during the 2nd SIM. Before the 2nd SIM terminates, the provider user logs out and his UIS expires immediately together with the OSP. The SBox1 keeps running and, at the 3rd SIM, the provider resumes the interactive session to complete their work and destroy the SBox. This time, however, because the user does not log out from HSM, his UIS lasts till the end of the 3rd SIM, also with no running sandbox (Diagram 2).



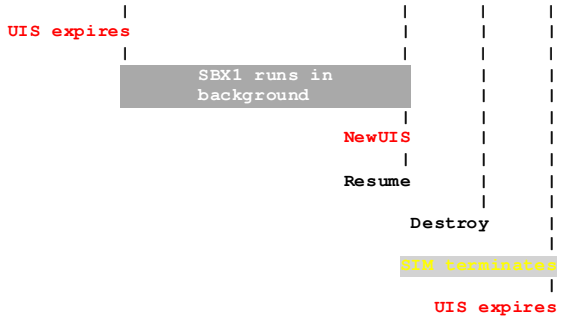
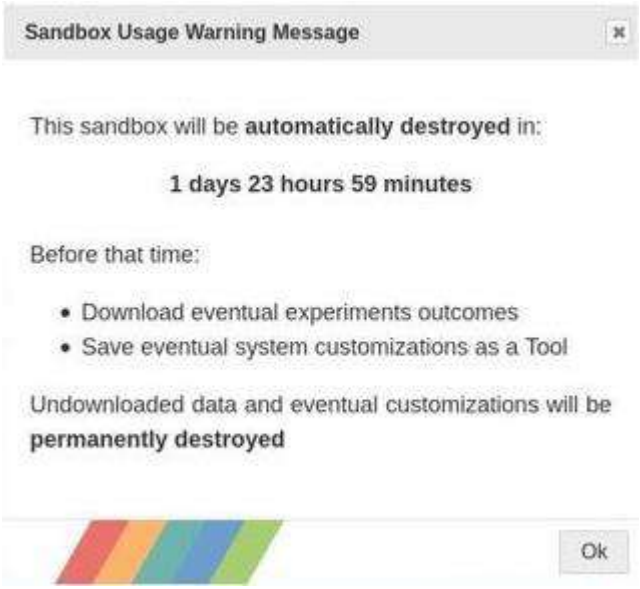


Diagram 2 - An example of how a provider user can take advantage of multiple SIMs and SBMs.

The Sandbox Instance - if not destroyed intentionally by its owner – will be un-recoverably destroyed by the system **at the end of the last SIM** granted to the owner's profile.

The sandbox owner is informed of the **overall duration of their sandbox**

- by a **dialog** appearing every time they open the Sandbox Viewer (again, the portion of HSM Web UI used to interact with the Sandbox).

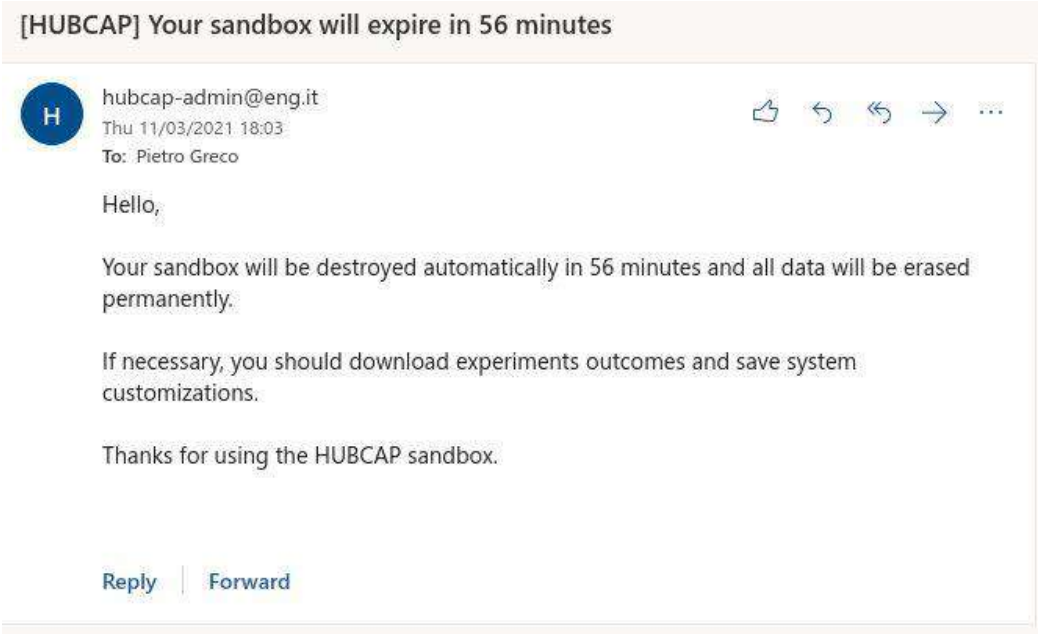


Dialog appearing when sandbox is accessed which shows the remaining sandbox lifetime

- by a **short message** always visible in the Sandbox Viewer

Sandbox remaining lifetime is also shown in the top bar of the Sandbox Viewer

Moreover, when the Sandbox is about to be destroyed by the system at the end of the last SIM, the owner will be informed by an **automatic system e-mail** sent about **one hour before** to allow saving any relevant data.



Email sent to inform about the incoming sandbox expiration

The feedback of a useful and professional internal assessment conducted by Frank Zeyda [VSI] on these new functionalities has allowed both a better tuning of them and their user's interface improving.

Annex 2 – Specification of Main Operating Systems provided in the HSM

In the table below are reported the specifications of the Operating Systems currently provided within the HSM.

Specifications of the HSM Virtual Machines for the different Operating Systems available

Operating System	vCPU(s)	vRAM (GB)	vDisk (GB)
Windows	4	8	16.0
Linux Desktop	2	4	5.0
Linux Terminal	1	1	2.5

Annex 3 – Platform HSM User Guide – v2

HUBCAP

Sandboxing Middleware

User Guide

Version 1.3

Sandboxing Middleware Release 2

Table of Contents

Document History

Version	Date	Author(s)	Description
V0.8	2020-07-27	Pietro Greco, Giuseppe Veneziano	
V0.9	2020-08-27	Pietro Greco, Giuseppe Veneziano	Replaced old images, added new images, extended tools provisioning paragraph
v1	2020-09-26	Pietro Greco, Giuseppe Veneziano	Updated §3.3. Feature overview table for Sandbox
v1.1	2021-01-28	Pietro Greco, Giuseppe Veneziano	Updated glossary (“OSes -> “Operating Systems”), modified paragraph on role of temp password in §4. picture of home page with full cart now replaces cart-only image. Added note on selected Models availability within the Sandbox Shared Storage in §5.1. Added notes
v1.2	2021-02-04	Pietro Greco, Giuseppe Veneziano	
v1.3	2021-05-26	Pietro Greco	Updated log in process description, included references to additional conditions for User Sessions Expiration and corresponding changes to UI, added note on one-session-only policy, updated images

1. Purpose of the document

The purpose of this document is to describe how to use the functionalities offered by the Hubcap Sandboxing Middleware (hereafter *HSM*).

2. Glossary

Word	**Description**
Sandboxing Middleware	Provides an environment where tools and models can be safely evaluated inside Sandboxes
Sandbox	Isolated set of running Tools, Operating Systems and Models featuring dedicated local network and shared storage
Sandbox Local Storage	A folder shared by all the Tools and Operating Systems in a Sandbox. Each sandbox has its own storage.
Sandbox Local Network	A TCP/IP Network available to all the Tools and Operating Systems in a Sandbox. Each sandbox has its own network.
Sandbox Viewer	Web page containing a Remote Viewer and a Control Panel through which it is possible to interact with the sandbox
Remote Viewer	HTML5 Remote Desktop Client
Collaboration Platform	Platform resulting from the combination of DIHIWARE portal and Sandboxing Middleware
Repository	List of instantiable Operating Systems, Tools or Models
Tool	CPS Software for Model Based Design which has been added to the Tools repository

3. Introduction

3.1 Sandboxing Middleware Overview

The Sandboxing Middleware provides an environment where several sandboxes can be executed concurrently without interfering with each other.

It is mainly composed of:

1. **Software Orchestrator:** needed to manage the environment and make it operational.
2. **Repositories:** containing Tools, Models, Operating Systems used as templates to create new running instances.

Different users can instantiate different sandboxes that can later be shared with other users.

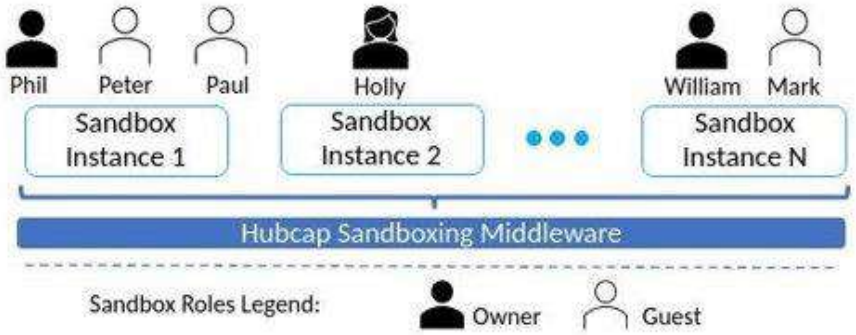


Figure 1 - Several sandboxes instantiated by different users running on the Hubcap Sandboxing Middleware

A **sandbox** is a set of running Tools, Operating Systems (OSes) and/or Models connected through a virtual, private and isolated network; the sandbox Tools and OSes also share a dedicated storage (Figure 2).

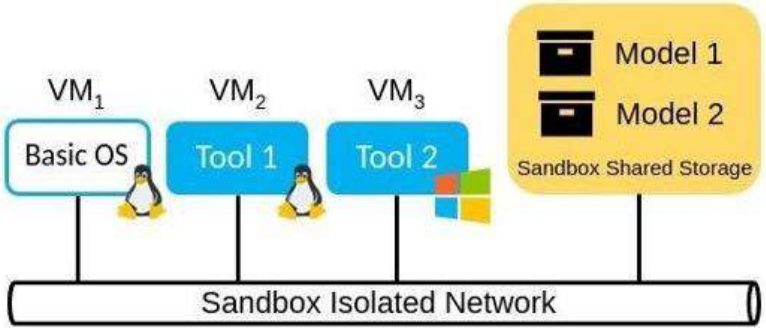


Figure 2 - Structure of a Sandbox Instance

Each Sandbox is instantiated dynamically by a platform user by picking up a set of items from the repositories:

- Operating Systems
- Tools
- Models

3.2 User Profiles and Roles

A user is characterized by a profile and one or more roles.

A **Profile** is assigned "statically" to a Collaboration Platform user and determines to which functionalities of the **Sandboxing Middleware** the user can access.

One or more **Roles** can be attached dynamically to a Sandboxing Middleware user and define the relationships between the user and the sandboxes thus determining to which functionalities of a **specific sandbox instance** a user can access.

Profiles

- **Consumer**: can instantiate sandboxes selecting tools and models from the Sandbox Middleware repositories (but cannot insert or destroy items from them).
- **Provider**: in addition to the Consumer's functionalities, a Provider can also:
    - Instantiate Operating Systems Virtual Machines on which to install their own tools,
    - Save these tools as well as upload models to the Sandboxing Middleware Repositories
    - Destroy the items they have previously added to the repositories.
- **Null**: cannot access the Sandboxing Middleware.

Roles

- **Owner**: is a Sandboxing Middleware user that instantiates a new sandbox. As an owner, they can:
    - Destroy their own sandbox
    - Share it with other Sandboxing Middleware users (who become their guests)
    - Upload or download local archives to/from the sandbox.
- **Guest**: is a Sandboxing Middleware user invited to access one or more sandboxes which they is not the owner of. Owner and Guests of the same sandbox can collaborate with each other sharing screen, mouse and keyboard of each Tool or Operating System within the Sandbox. A guest cannot upload or download archives, destroy or share the host sandbox.

So, a Collaboration Platform user

- has one static Sandboxing profile (Consumer, Provider, Null)

and, if they can access the Sandboxing Middleware, they

- can be Owner of **one only** sandbox instance at the time,
- can be Guest of **many** sandbox owners.

3.3 Features Overview

The following table outlines the main functionalities made available by the Sandboxing Middleware as well as the combinations of users' profiles and roles for which they are intended.

Feature	Owner Role		Guest Role
	Provider Profile	Consumer Profile	Whatever Profile
Consecutive SIMs	**3**	**1**	-
Access to remote viewer	X	X	X
Upload Archive	X	X	-
Download Archive	X	X	-
Invite Guests	X	X	-
Destroy Sandbox	X	X	-
Select Tool	X	X	-
Select Model	X	X	-
Select Operating System	X	-	-
Save a New Tool	X	-	-
Save New Tool Version	X (their own)	-	-
Upload New Model	X	-	-
Delete Repository Item	X (their own)	-	-

The **Number of Consecutive Interactive Sessions** represents - with some degree of approximation - the **maximum number of days across which the Sandbox is preserved.** When a user has a profile with this parameter greater than 1, they can resume working the next day on the same sandbox instance left the day before.

4. Accessing the Sandboxing Middleware

To access the Sandboxing Middleware, first access the portal at the following URL with your credentials:

https://hubcap-portal.eng.it/

then click on the "Sandbox Environment" button available in the sidebar:

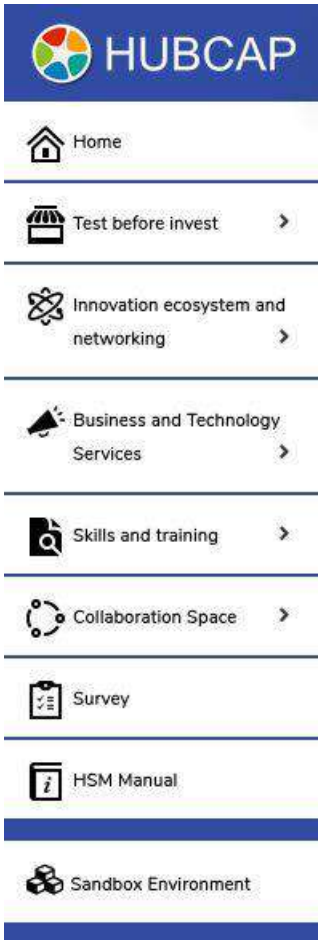


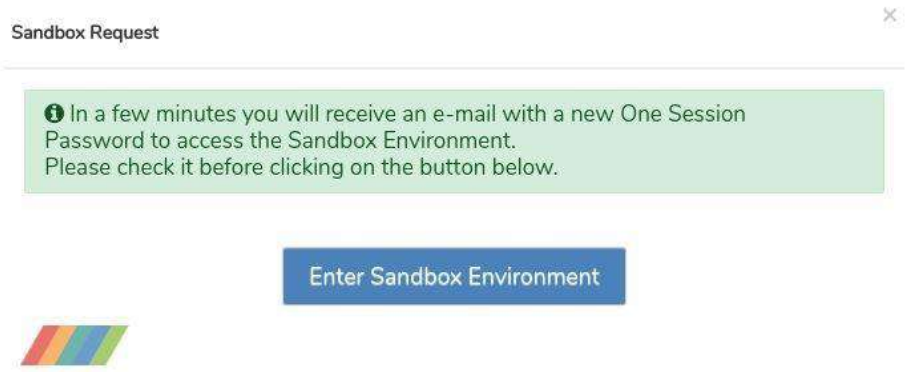
Figure 3- Collaboration Portal Main Sidebar Menu

Note: Currently there are two modes of operation for the Sandboxing Middleware:

- **System Interactive Mode**: daily time slot within which users **can log in** and use the middleware interactively
- **System Batch Mode**: entered when the daily time slot is over. The users **cannot log in**, but sandboxes instantiated during the Interactive Mode will keep on running

After clicking on the “Sandbox Environment” button, a dialog will inform the user which one of the two modes - Batch or Interactive - is active. If the Interactive mode is on, the user will be able to log in and the dialog will say whether a new temporary password (One Session Password) has been sent to the user’s email address or the one previously sent is still valid.

To access the HSM the user then clicks on the confirmation button included in the same dialog - “Enter Sandbox Environment”. After clicking, the HSM welcome page will be opened in a new browser tab.



Use the received credentials to access the Sandboxing Middleware web interface.

	Note: some browsers (e.g., Chrome) block the Sandboxing Middleware welcome page pop-up. Unlock it or click on the alternative link to proceed.

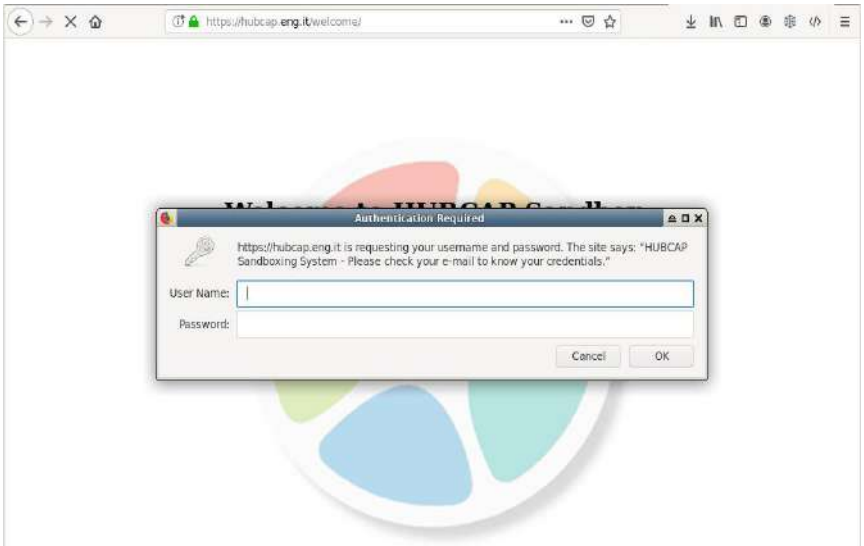


Figure 4 - Sandboxing Middleware Login Page

The temporary password is only valid for a predetermined period of time and expires:

- when the user clicks on “Logout”

- at the end of the *daily timeslot* (called **System Interactive Mode**)
- it has been about 10 minutes since an HSM tab browser closure
- it has been about 10 minutes since a browser closure
- the user remains idle on the HSM web interface for more than about 15 minutes

Note: when the user remains idle on the HSM web interface for more than about 15 minutes, the Sandboxing Middleware informs the user about the incoming session invalidation through the following pop-up dialog:

At that point, by clicking on either the close ("x") or Ok button the user will be able to reset the inactivity detection mechanism and continue to use the middleware.

During the System Interactive Mode, if the user logs out or the session (and password) is automatically invalidated, he/she can request a new password by accessing the HSM again from the portal.

The session expiration time is shown at the top of the HSM home page:

Figure 5- Sandboxing Middleware session expiration shown in the HSM home page.

Note: if the user does not click on Logout but instead only closes the tab or the browser, the session will remain active for about 10 minutes. If the browser has been closed, this means – for security reasons - the user will not be able to access until the session expires after 10 minutes.

Note on the One-Session-Only policy: the browser closure case described in the previous note is a consequence of the One-Session-Only security policy, according to which a HSM user can only have a single active session at a given time. If a user is currently logged in or has an active unexpired user session, any attackers trying to impersonate them will not be able to log in from another computer.

The following sections describe how to take full advantage of the capabilities offered by the middleware.

5. Tools and Models Evaluation

This chapter describes how to carry out evaluation experiments with Tools and Models from a **Consumer user perspective**.

	Note: The Provider users **are still required to know** the content of this chapter (chapter 5).

Once logged in to the Sandboxing Middleware, the user is prompted with the home page (Figure 6):

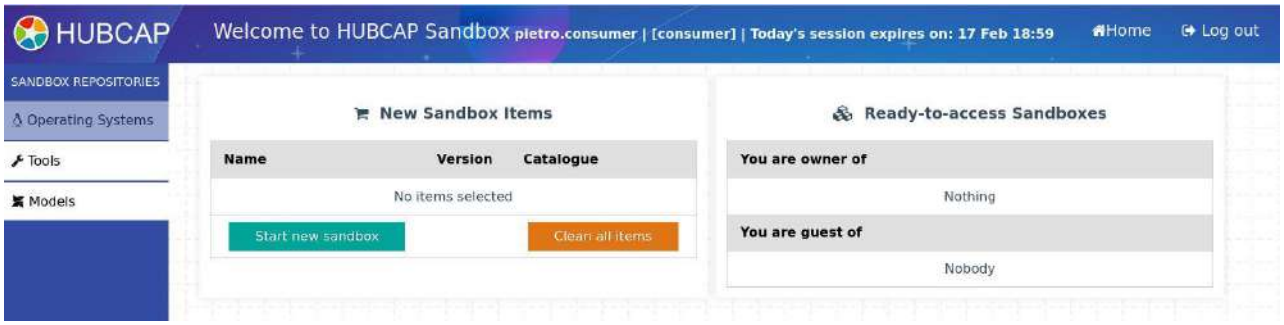


Figure 6 - Sandboxing Middleware Home Page.

From here, evaluation experiments can be started.

5.1 Selecting Tools and Models

As first step, the user selects the *Tools* and/or *Models* he wants to evaluate from the corresponding repositories. These are accessible from the left sidebar in the home page (Figure 7):

Figure 7- Sandbox Repositories on Sandboxing Middleware Sidebar Menu

Once the repository of interest is accessed, the specific item (*Tool* or *Model*) can be selected by clicking on the corresponding Add to sandbox button (*Figure 8*):

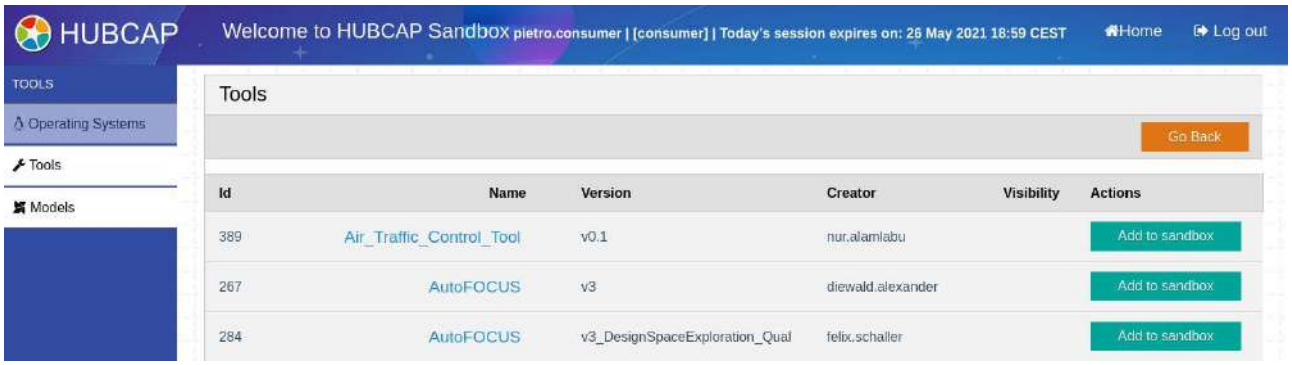


Figure 8 - Tools Repository

It is also possible to view the details of the item by clicking on the item name:

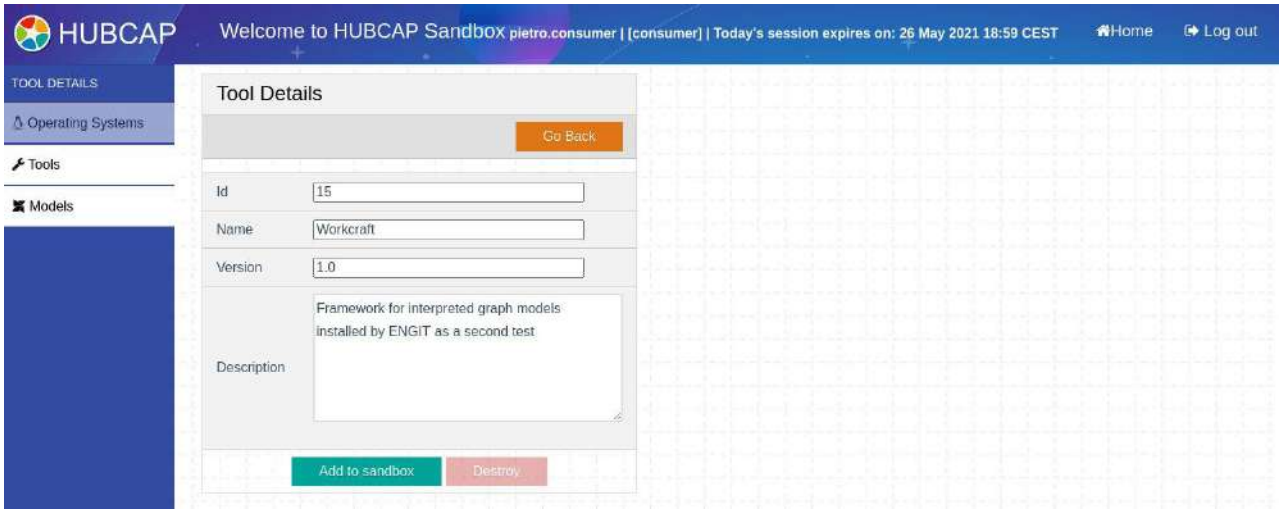


Figure 9 - Tool details page

Note: the user can select the same Tool multiple times. Multiple instances of it will run simultaneously in the Sandbox.

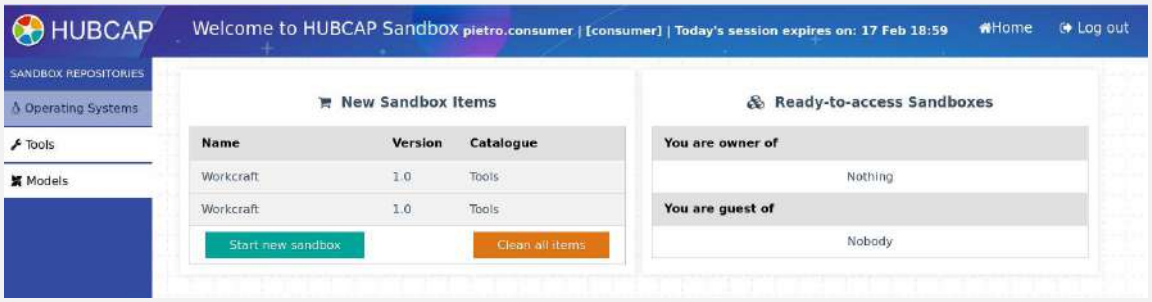


Figure 10 - Same Tool selected twice.

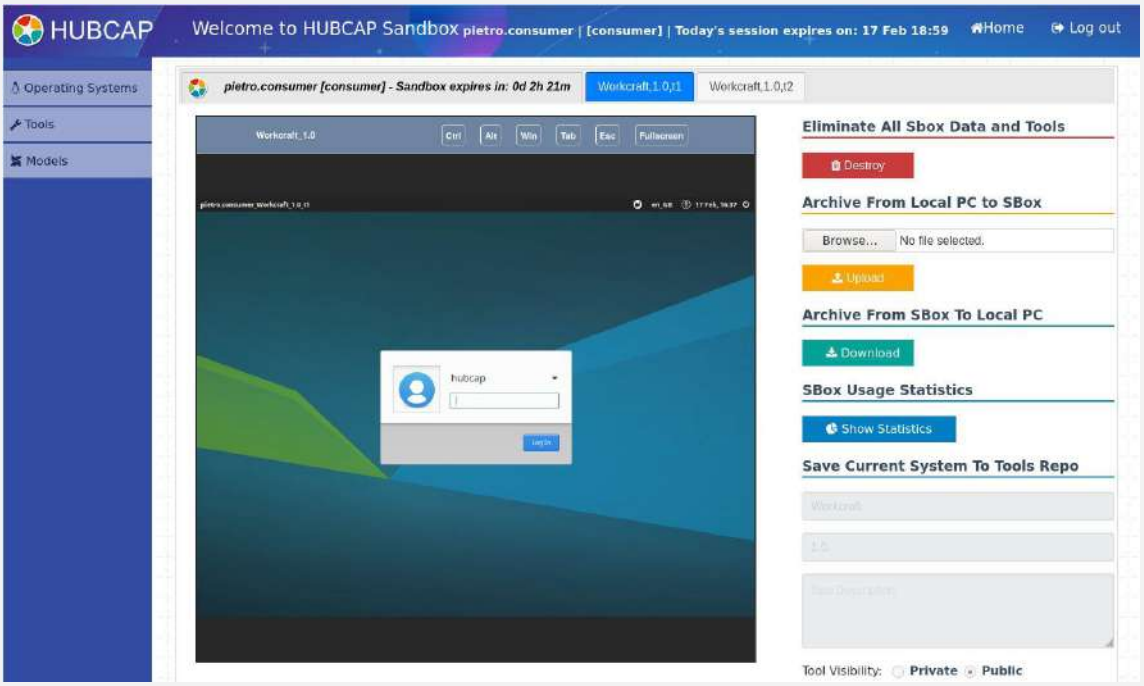


Figure 11 - Two instances of the same Tool running in the Sandbox.

In case a user also wants to select or view the details of a model, they can repeat the just described procedure referring to the Models Repository.

A recap of the *Tools* and *Models* added to the sandbox is available on the home page under the "New Sandbox Items" section (Figure 12):

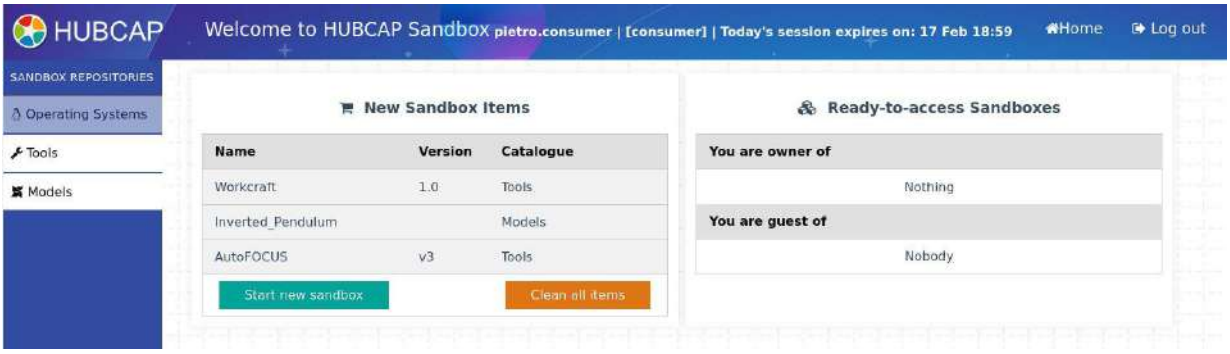


Figure 12 – Home page featuring a cart (New Sandbox Items section) with two Tools and one Model.

In case the user wants to exclude or replace an item from the cart, he clicks on Clean all items to reset the cart.

> **Note:** The selected Models, as stated later in §5.3.2, will be available to any Tool or Operating Systems running in the Sandbox from the `/nfs/toolsdata/<username>` folder.

5.2 Starting the Sandbox

Once the items needed are selected, the user can launch the new sandbox by clicking on Start new sandbox, available in the "New Sandbox Items" section of the home page. The Sandboxing Middleware will start to instantiate the desired *Tools* and *Models* within a new Sandbox.

> **Note**: It is possible to instantiate **only one** Sandbox per time. If the “Start new Sandbox” button is disabled, you are already Owner of a running sandbox. In that case, you can access your existing sandbox by clicking on the corresponding button in the “You are owner of” section of the HSM Home Page.

5.2.1 Sandbox Lifetime

Every Sandbox has a lifetime after which it *is automatically destroyed,* and depends on the number of consecutive Sandbox Interactive sessions (SIMs) associated with the specific user profile (see table in section 3.3 above for further information)

One hour before the expiration, the Sandbox owner receives a reminder email advising them to save any unsaved data.

The Sandbox lifetime is shown in the top area of the viewer that appears once the sandbox instantiation phase is over (Figure 13):

While the sandbox is alive and during the Sandbox Interactive Mode, the user is free to leave and access it again later.

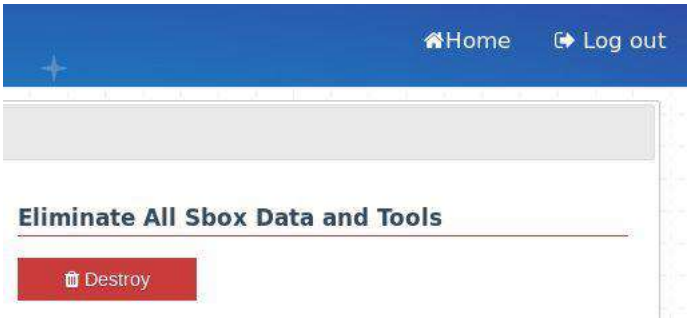


Figure 13 - Top right area of Sandbox Viewer featuring three buttons: Home, Log Out and Destroy

In particular, from the viewer, the user can click on "Home" to get back to the Home page - from where he can access other users' sandboxes - or on “Log out”. In both cases, and in the second (“Log out”) only after having obtained a new temporary password by accessing again from the portal, he can access his running sandbox again from the “Ready-to-access Sandboxes” section, “You are owner of” subsection, in the home page:

Figure 14- Ready-to-access Sandboxes section with detail on accessible Sandbox owned by a logged user.

5.3 Use of the Sandbox

Once the instantiation process is complete, the Sandbox Viewer will be prompted (Figure 15):

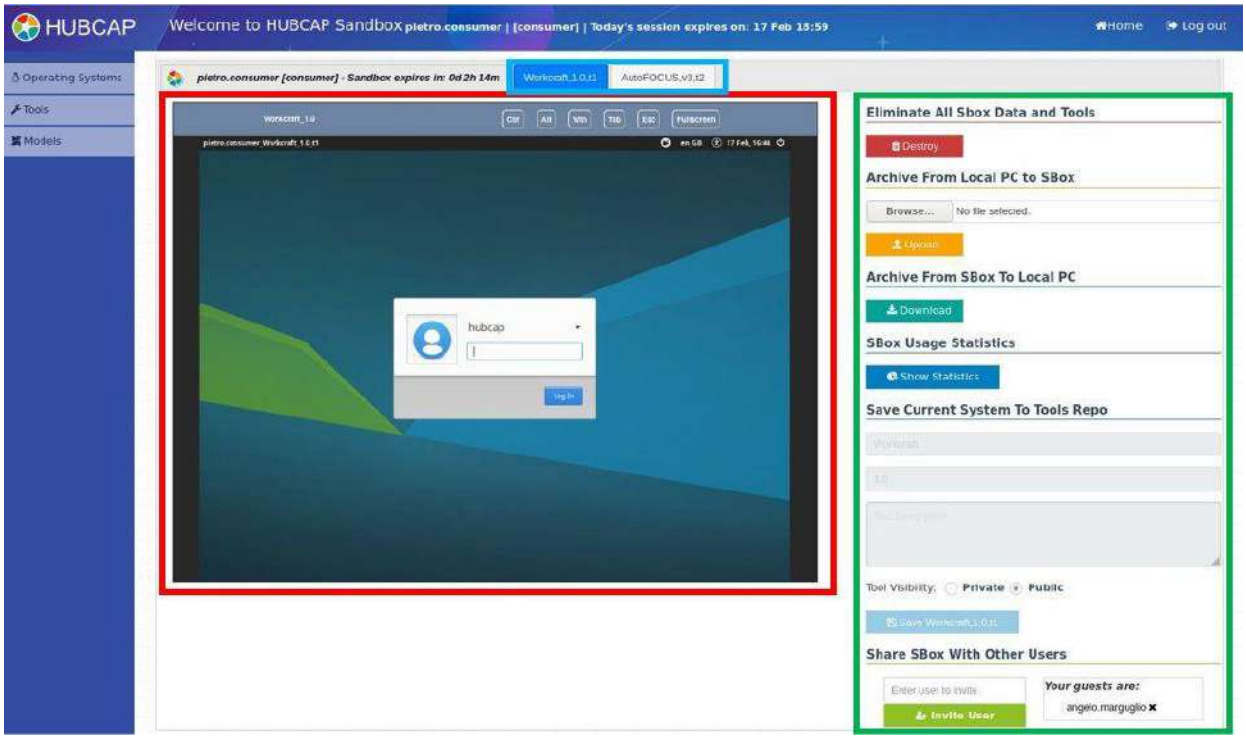


Figure 15 - Sandbox Viewer with three main components highlighted.

The **Sandbox Viewer** is composed of three principal areas:

- Remote Viewer (red area)
- Tool Selector (blue area)

- Control Panel (green area)

The **Remote Viewer** provides the user with access to the user interface of the Operating System on which the Tool has been installed and therefore to the Tool itself. It also features a set of buttons (Ctrl, Alt, Win, Tab, Esc) - some of which toggle buttons (Ctrl, Alt, Win) - that can be used to send special combinations of keys to the operating system. In addition, the Fullscreen button allows the user to switch to Fullscreen mode.

	Note on Fullscreen mode: once Fullscreen mode is on, for a better user experience, you might want to change the screen resolution from the Operating System in the sandbox.

	Note on OS/Tool access: the Operating System asks for a **password**. The default one, if not changed by the Tool Provider, is *hubcap*

From the **Tool selector**

Figure 16 - Tool selector in the top area of Sandbox Viewer.

the user can switch from a Tool to another one by clicking on the corresponding tab. The Remote Viewer will be updated accordingly.

Finally, the **Control Panel:**

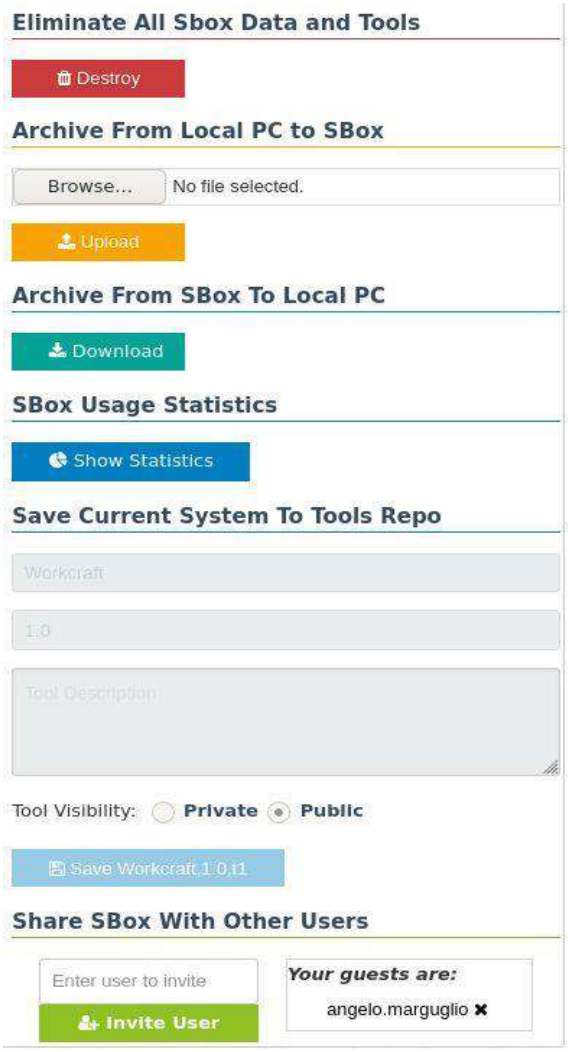


Figure 17 - The Control Panel is located on the right of the Sandbox Viewer.

gives access to a set of functionalities – described in what follows - available depending on the combination of role and profile assigned to the user.

5.3.1 Interaction among Tools

Different Tools running within the same Sandbox can interact with each other through:

1. the **Sandbox Local Storage**, which is a folder they **share**, or
2. the **Sandbox Local Network,** referring to each other via their hostnames or IP addresses.

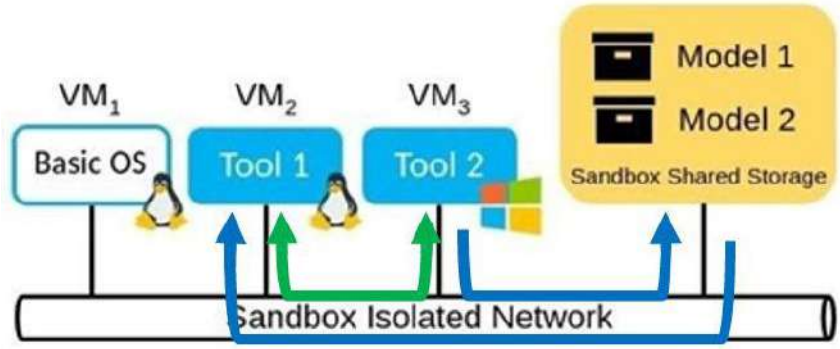


Figure 18- The Isolated Network and Storage allow the tools of the Sandbox to interact with each other.

5.3.1.1 Interaction through the Sandbox Local Storage

Files added to the `/nfs/toolsdata` shared folder can be accessed from any Tools through the default File Manager offered by the operating system on which they are installed (Figure 19).

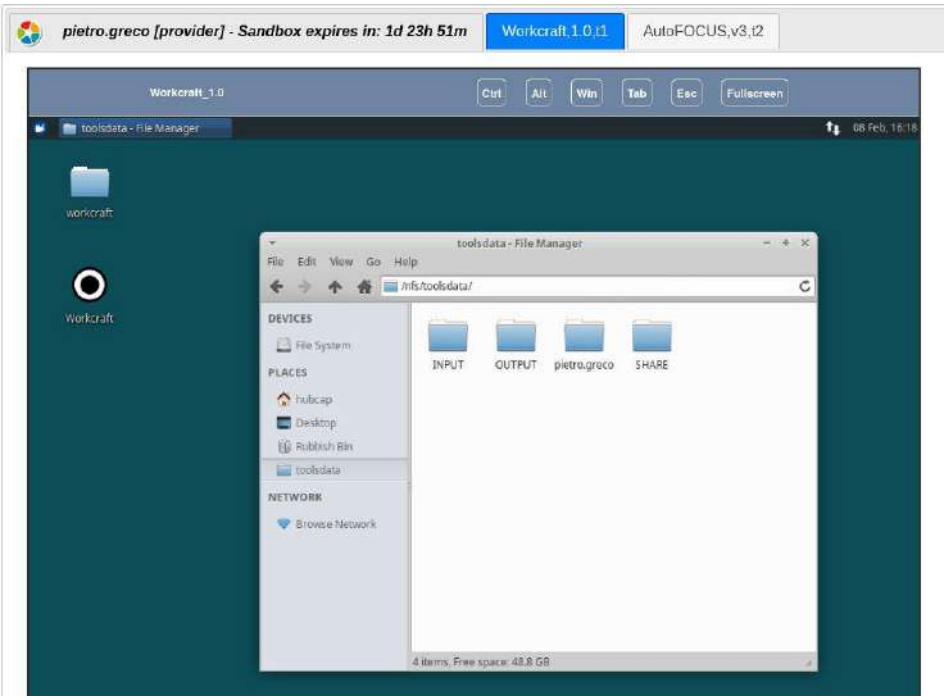


Figure 19 - File manager of Operating System running inside the Sandbox opened on the Sandbox Local Storage (/nfs/toolsdata)

Note: The Sandbox Shared Storage cannot be accessed from other sandboxes.

5.3.1.2 Interaction through the Sandbox Local Network

The Tools can also interact through the **Sandbox Local Network** referring to each other via their hostnames or IP addresses.

To get the hostname or the IP address of a Tool instance you can open, from it, a terminal and use the following commands:

Information / OS	**Linux**	**Windows**
Hostname	`hostname`	`ipconfig /all`
IP address	`hostname -I`	

Note: The Sandbox Local Network cannot be accessed from other sandboxes.

5.3.2 Accessing the selected Models

The Models selected and added to the Sandbox during its composition are available in the **Sandbox Local Storage** under `/nfs/toolsdata/<username>.` This folder, being under `/nfs/toolsdata`, is accessible from any Tool of the same Sandbox.

5.3.3 External Interaction with the Sandbox Local Storage

Note on *Profile* and *Role*: This and the following paragraphs describe those functionalities relevant for the evaluation experiments and available to users with Owner *role* (those who have instantiated the sandbox) and Consumer or Provider *profile*.

5.3.3.1 Upload to the Sandbox

To **upload** an archive from his local computer to the Sandbox, the *Sandbox Owner* interacts with the following widget (Figure 20):

Figure 20 - Upload area from which it is possible to upload an archive into the sandbox. The upload content will be available inside the INPUT folder of the Sandbox Local Storage

In particular, he selects the archive from his local computer by using the "Browse ..."/"Choose File" button and then clicks on "Upload".

The uploaded archive is now available in the *Sandbox Local Storage* under `/nfs/toolsdata/INPUT`, and from there again available to all the Tools in the Sandbox.

	Note: It is not possible to access files in the Sandbox Local Storage (/nfs/toolsdata) from other sandboxes

	Tip: by uploading an archive the user can upload project files, license keys, manuals and in general data he/she wants to use within the sandbox but does not want to share with other sandboxes.

5.3.3.1 Download from the Sandbox

Furthermore, the Sandbox Owner can **download** the results of the evaluation experiments (CPS software output, notes, reports, etc.) by adding them to the `/nfs/toolsdata/OUTPUT` folder and then clicking on "Download" (Figure 21, available in the Control Panel):

Figure 21 – Clicking this button, it is possible to download an archive containing the content of the OUTPUT folder (available in the Sandbox Local Storage)

The user's browser will open a download dialog from which an archive containing the experiment outcomes can be downloaded (Figure 22).

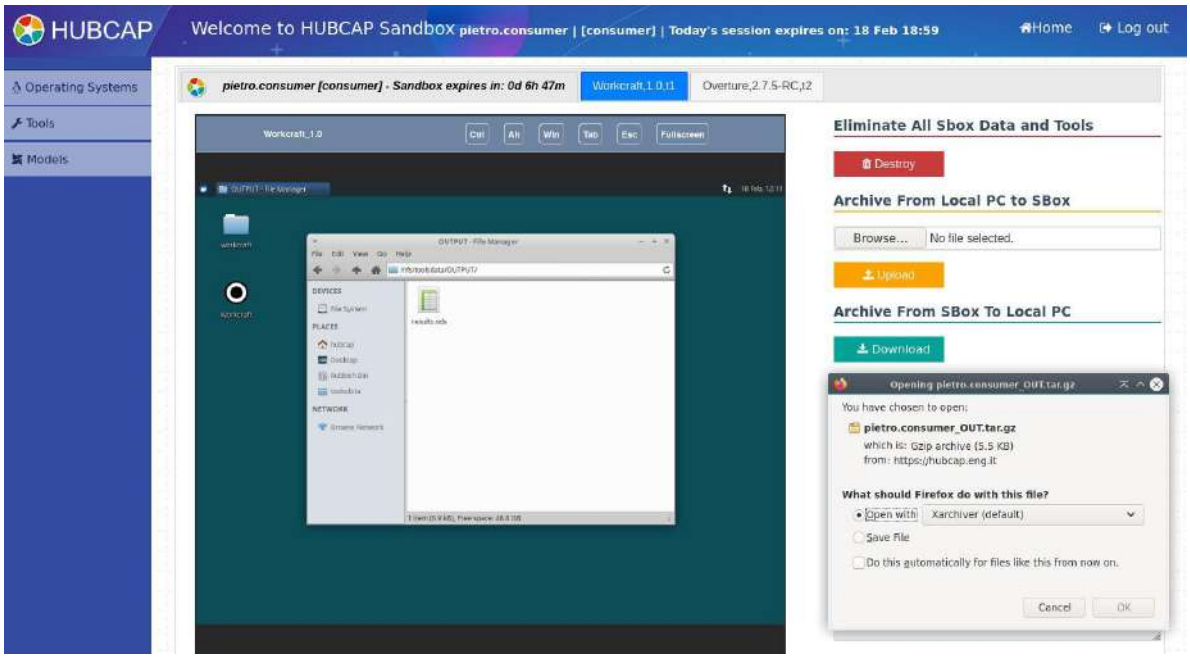


Figure 22 - Download dialog prompted after clicking on Download.

5.3.4 Visualizing Sandbox Usage Statistics

Clicking on “Show Statistics” in the control panel:

Figure 23 - The Show Statistics button is available in the control panel

the user visualizes, for each Tool and for the Shared Storage, information about the resources they have been consuming:

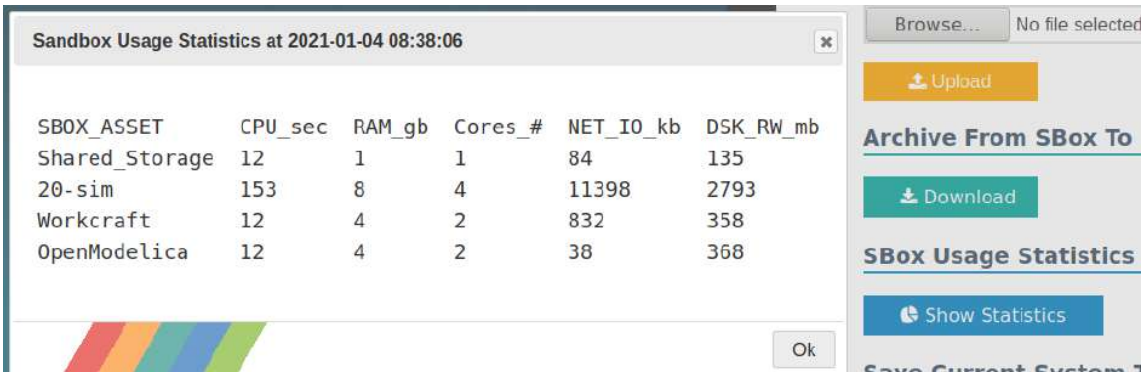


Figure 24 - After clicking on the Show Statistics button, a dialog will show the resources the sandbox items have been consuming

A final report on resources consumption will also be shown after the destruction of the sandbox.

5.3.5 Sandbox Sharing

Through the widget depicted below (Figure 25), the Sandbox Owner can choose with which users (guests) to share his sandbox.

Figure 25 - Sandbox Sharing Panel (in Control Panel) from which guests can be invited.

To do so, the Sandbox Owner uses the autocomplete search bar and the "Invite User" button (in Figure 25) to select the user to invite as a guest.

At that point, the **Guest user** - logged in to the Sandboxing Middleware - can access the sandbox by clicking on the corresponding button available in the home page - "Ready-to-access Sandboxes" section, "You are guest of" subsection - named with the sandbox owner's name (Figure 26).

Figure 26 - Ready-to-access Sandboxes section showing the sandbox owned by another user.

After clicking on that button, the Sandbox Viewer (*Control Panel excluded*) will be prompted to the Guest.

To stop sharing the sandbox with a specific user, the sandbox's owner clicks on the button named with the guest's name in the "Your guests are" section (Figure 25).

5.3.5.1 Collaboration Scenarios within the Sandbox

Multiple users connected to the **same Sandbox** can work simultaneously on the **different Tools running in the Sandbox.**

Furthermore, the users connected to the Sandbox can also access the **same Tool**, sharing its screen; **only one virtual keyboard** and **one virtual mouse** are **shared** for collaboration among all the users interacting with the same tool instance.

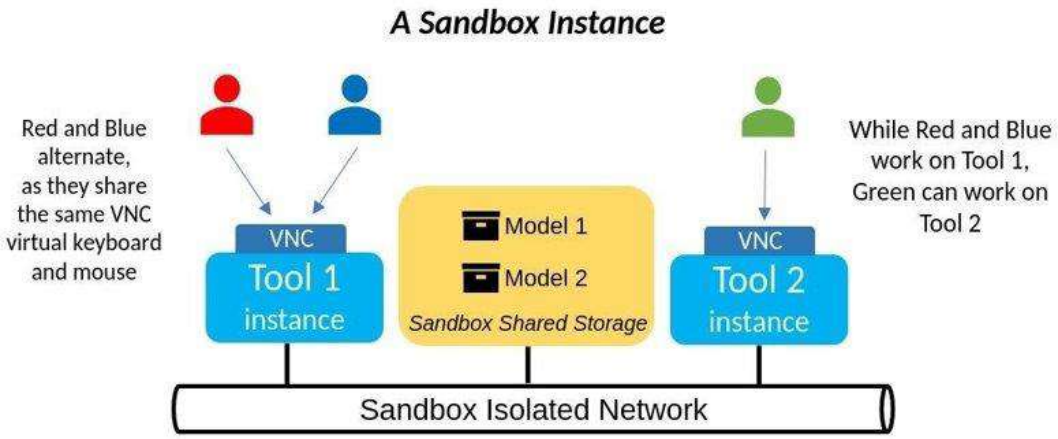


Figure 27 - Typical collaboration scenario within the Sandbox.

	Tip: during the composition of the Sandbox the user can select the same Tool multiple times. This means that the different users of the Sandbox will be able to work on these different instances simultaneously and independently.
	Note on private tools: if the Sandbox contains sandbox owner's private tools, guests will be able to interact with them until the sandbox is destroyed.

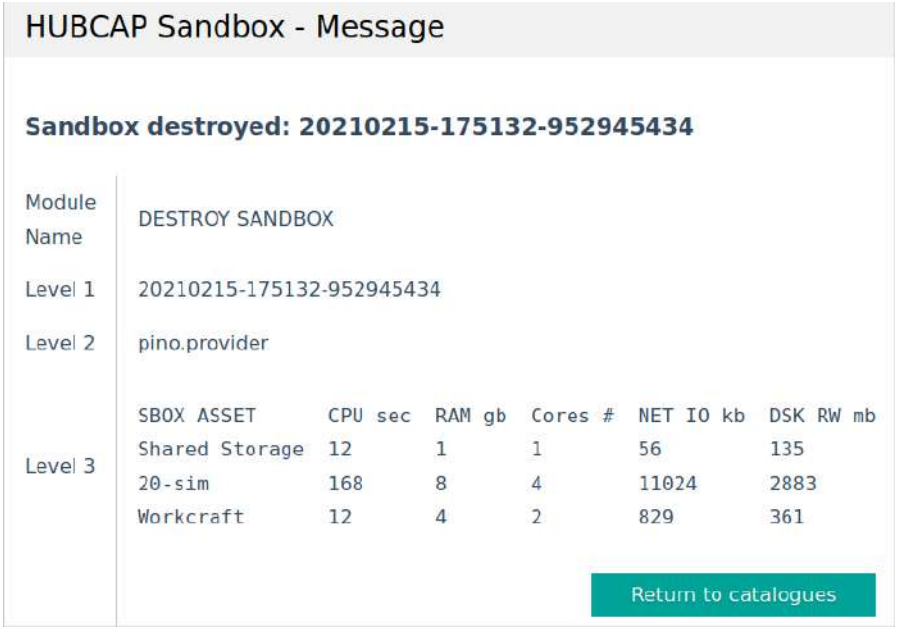
5.3.6 Destroying the Sandbox

Warning: files and any kind of data available in the Sandbox (composing Tools and Local Storage) are deleted with the destruction of the Sandbox and **are not recoverable**. Be careful!

To destroy the Sandbox and all its content click on “Destroy”. Guest users will be disconnected.

Note: users who do not need the Sandbox anymore should always destroy it.

Upon Sandbox destruction the HSM shows a report on resources consumption:



6. Provisioning

In addition to the basic functionalities illustrated in the previous chapter, a Provider user also has access to those intended for bringing new *Tools* and *Models* to the Sandboxing Middleware Repositories.

> **Note**: The Provider users **are required to know** the content of Chapter 5. Tools and Models Evaluation

6.1 Tools Provisioning

A Provider can either:

1. Provision their CPS software by installing it on an empty operating system – provided by the Sandboxing Middleware itself, or
2. Create a new version of a Tool *they had previously added* to the middleware.

> **Note**: A Provider cannot modify Tools provisioned by other providers

The following steps illustrate how to provision a new piece of CPS Software starting from an empty operating system. The same procedure applies in case the Provider wants to **create a new version of one of his/her own tools**; instead of selecting empty operating systems they will have to select their Tools.

> **Tip**: for a Provider who wants to create a new version of one of his Tools it might be useful to instantiate the Tool twice. In this way, he will have the possibility of modifying only one instance and comparing its performance with that of the unmodified one.

1. To add new *Tools* to the Tools Repository, as first step the Provider *selects* from the Operating Systems Repository (Home Page -> Left Sidebar -> Operating Systems) (Figure 28), *as many **Operating Systems** as the Tools he wants to provision*.

> **Tip**: In addition to the operating systems, the Provider might also want to select already existing Tools and Models. In that way it will be possible to evaluate, within Sandbox, how well the to-be-saved/being-provisioned Tools interact with other Tools and Models.

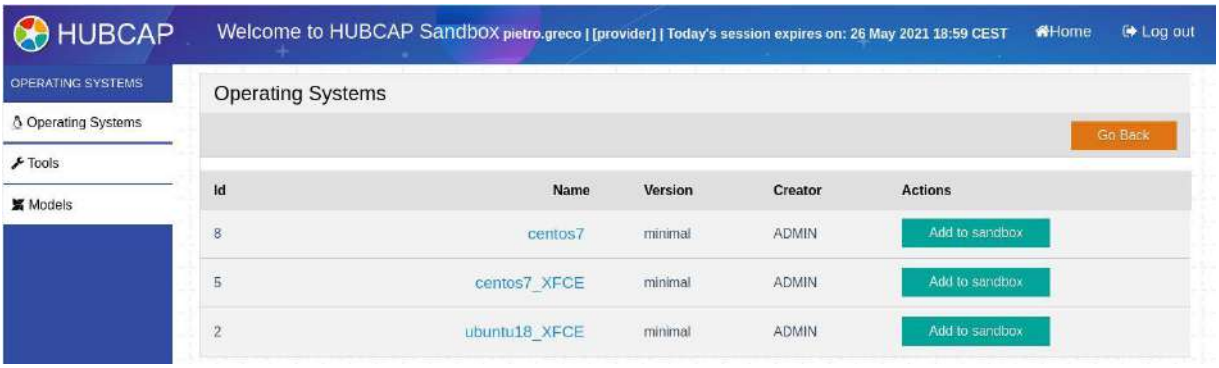


Figure 28 *- Operating Systems repository*

	Reminder: An Operating System is a VM on which an empty OS has been installed. You customize this VM by installing your CPS software on it thus creating the Tool (i.e., a VM with the CPS software installed)

2. He then clicks on  in the home page to instantiate the new sandbox.

	Note: If the “Start new sandbox” button is disabled, you already have a running sandbox. A user can only have one running sandbox per time and there is no exception for the provisioning phase. If needed, destroy the running sandbox as indicated in the previous chapter.

Once the Sandbox instantiation process is complete, the resulting Sandbox Viewer will show - in the top area - *as many tabs as the selected Operating Systems (Figure 29)*.

Figure 29 - Sandbox Viewer top area detail showing two tabs, one for each Base Operating System running in the Sandbox

3. The Provider as next step selects through the Tool Selector the Operating System he/she wants to customise and installs on it his/her CPS software.

	Tip: For the installation purposes, the Provider can leverage all the functionalities described in the “Tools and Models Evaluation" chapter 5. So, for instance, he might want to use the upload archive

functionality to upload installation scripts, license keys or manuals. Or, if needed, obtain the same content by accessing - from the Operating System within the Sandbox - the public internet.

As an example, in the following steps we will show how to install Eclipse and how to add it to the Sandboxing Middleware Tools Repository.

4. The Provider clicks on the tab corresponding to the instance of Operating System he wants to customize and accesses the OS by entering the OS credentials (default password: *hubcap*) (Figure 30):

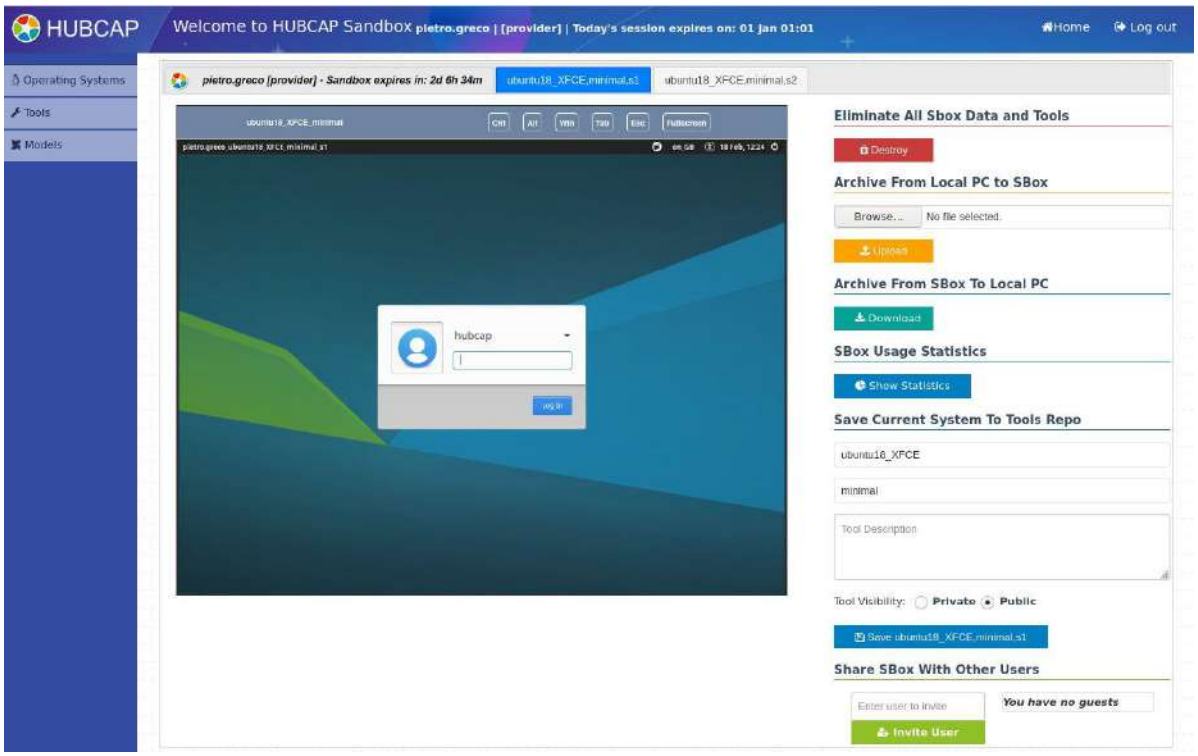


Figure 30- Operating System Login Dialog

5. The provider accesses the Internet from within the *Operating System* instance to download the Eclipse installer (Figure 31):

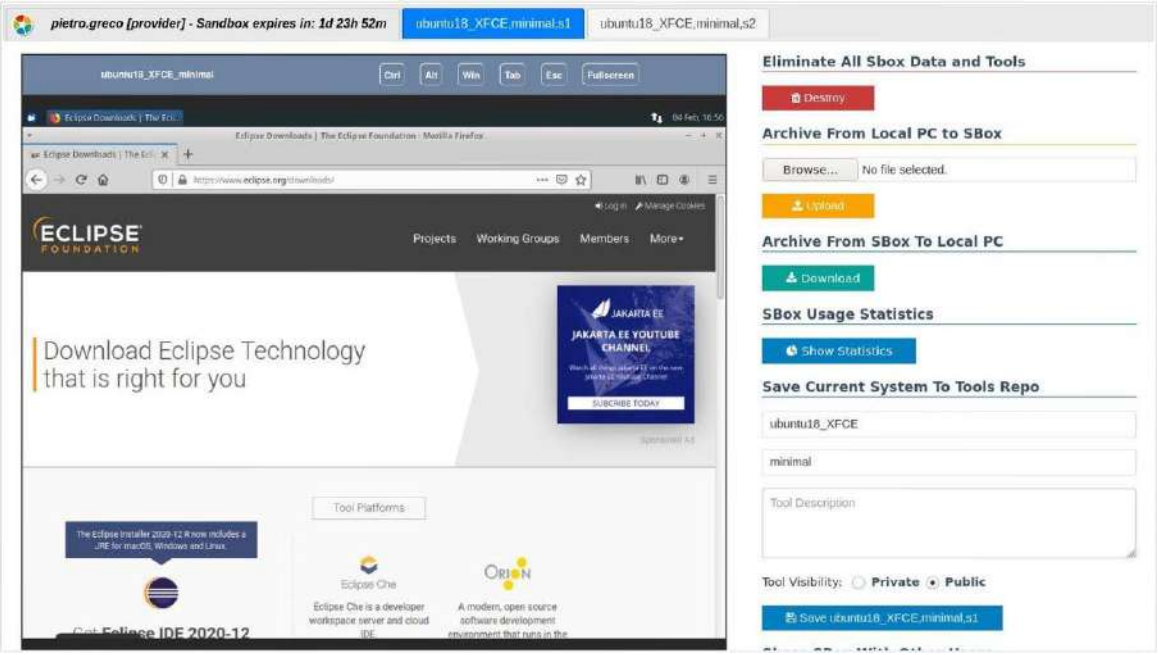


Figure 31 - Browser open on an Operating System running inside the sandbox. The Eclipse download page is shown.

6. The provider also downloads and installs the necessary dependencies. In this case, the Java runtime (Figure 32):

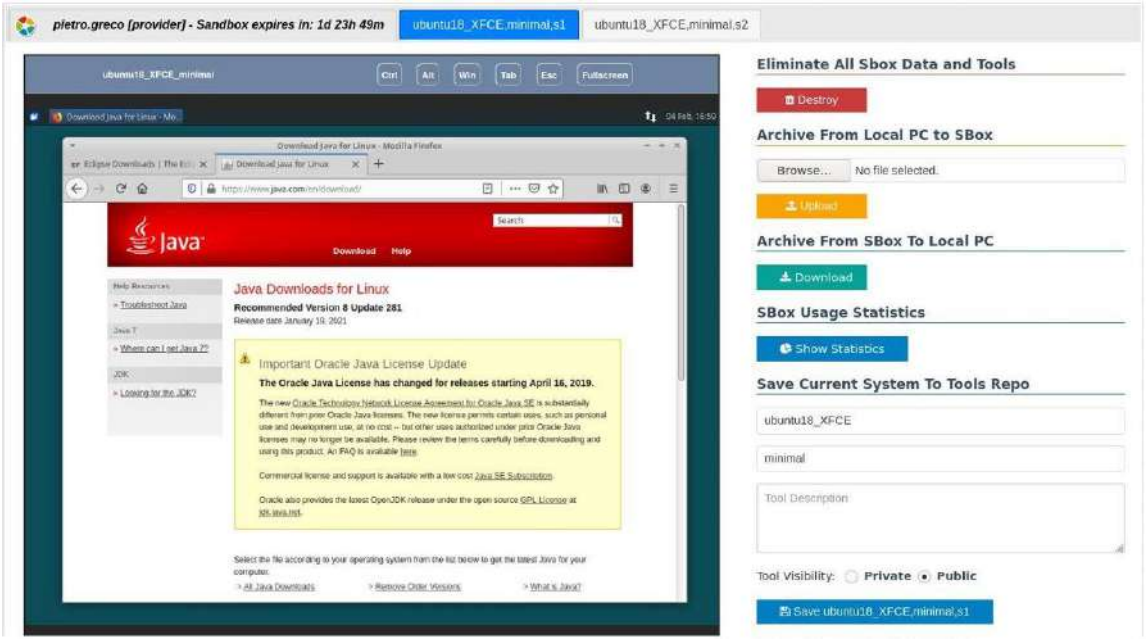


Figure 32 - Downloading dependencies needed by Eclipse.

7. The provider installs Eclipse by using its installer (Figure 33):

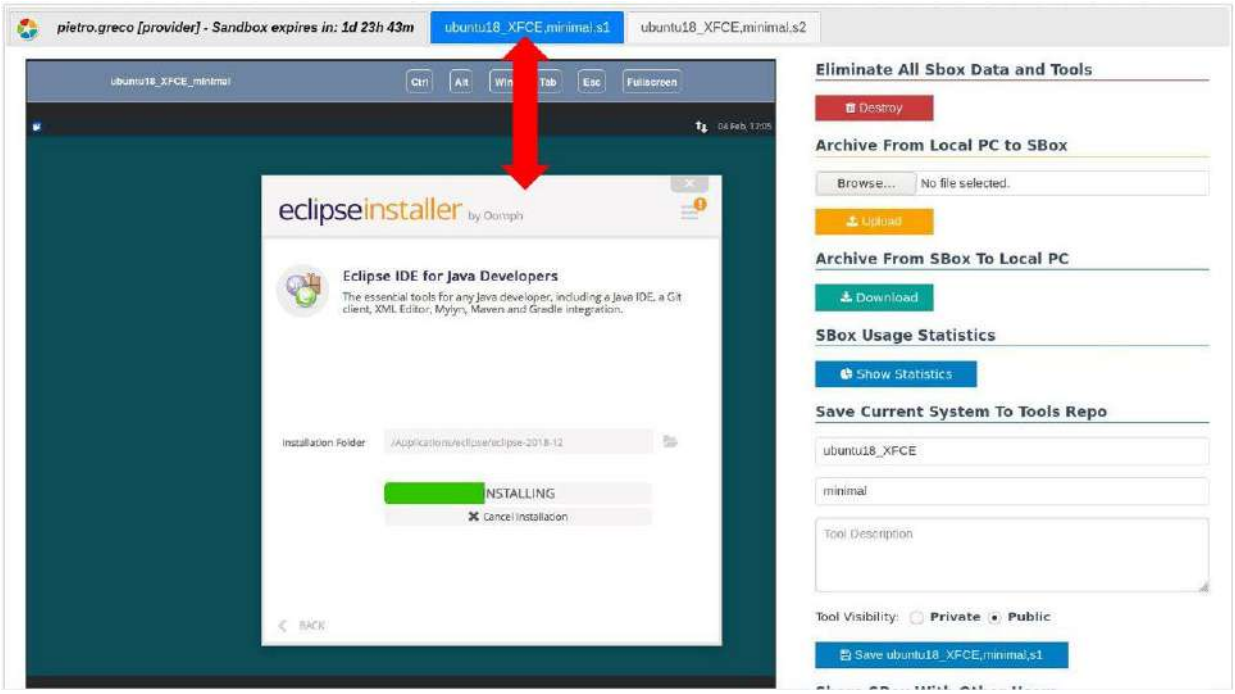


Figure 33- Sample software being installed on the selected Operating System (notice selected tab)

	Note: On each *Operating System* you should install only one Tool (i.e., a one-to-one relationship between a *Tool* and an *Operating System* should exist).

8. The provider configures and tests the installed tool (Figure 34):

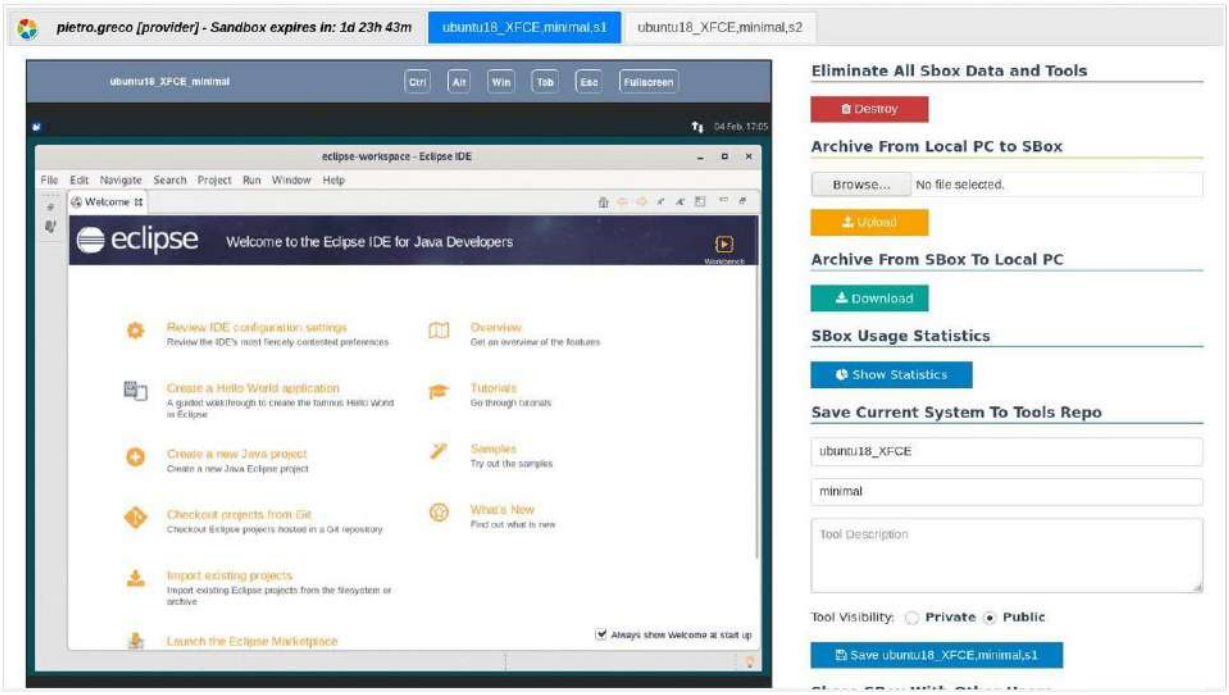


Figure 34 - Eclipse installed and ready to be tested.

Once the tool is completely installed, configured and tested, the provider **saves it** (next steps, 9, 10, 11) thus adding it to the Sandboxing Middleware Tools Repository.

	Tip: The status of the Tool instances at the time of saving is the one that Consumer users will find every time they instantiate it from the repository. Therefore, before saving, pay attention to UI language, screen resolution, default views, access passwords and so on. Also, pay attention to any sensitive data you might have stored in the operating systems during the installation.

	Tip: To save disk space and reduce tool instantiation time, delete any temporary files you have used during the installation, and not needed for the evaluation experiments.

9. To save the Tool, the provider clicks on the tab corresponding to the just customized operating system (Figure 35):

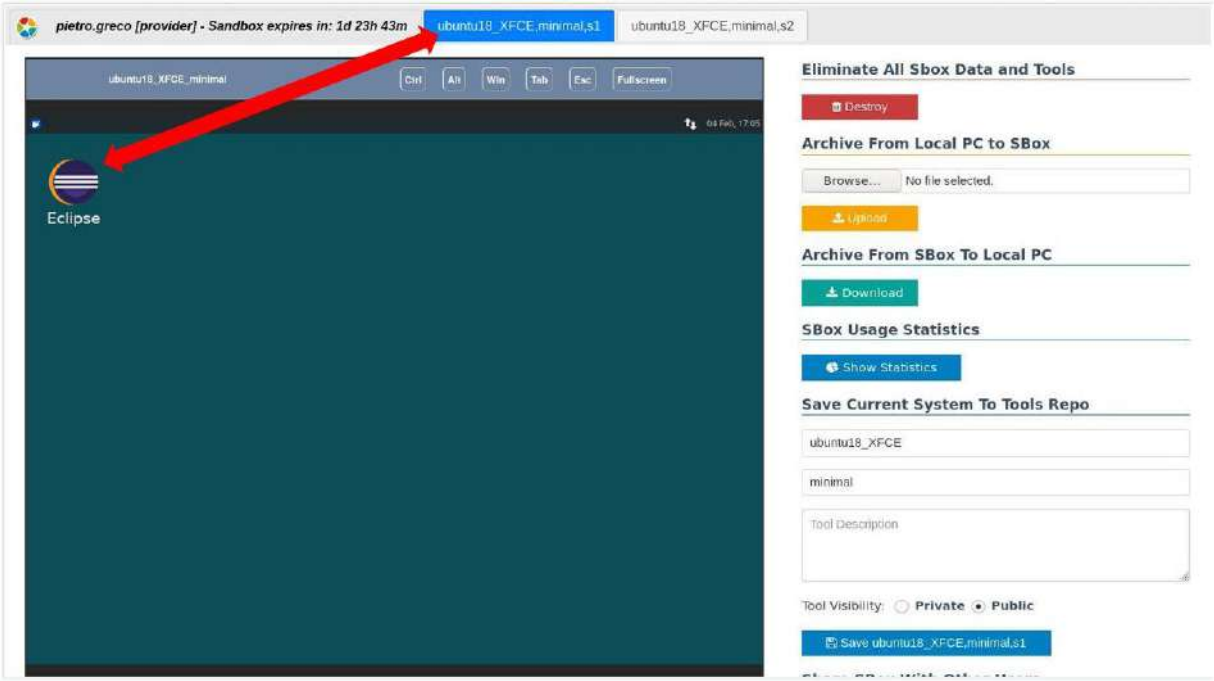


Figure 35 - User has just selected the tab corresponding to the customized Operating System he wants to save

10. The provider enters tool name, version, description and selects the appropriate Tool visibility level through the following form available on the right in the Control Panel (Figure 36):

Figure 36 - Save as a tool form available in the Sandbox Viewer Control Panel

If the visibility is set to "Private", the *Tool* will be shown in the Tools repository only to the Provider user who has saved it. Providers can later change the visibility settings of their tools from the Tools repository (*Figure 37*):

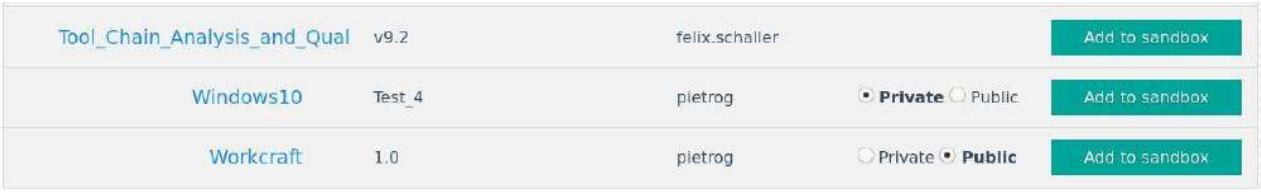


Figure 37 - From the list of Tools in the Tools Repository the tool owner can change the visibility settings of his/her Tools.

If the visibility is set to "Public", the Tool will be shown to all the users of the Sandboxing Middleware accessing the Tools Repository.

> **Note:** if a user shares a Sandbox containing instances of their own *private* tools, the guests can use these private tools as long as the Sandbox is running.

11. When the information about the tool has been entered, the provider clicks on "Save" to save and add the tool to the Tools repository.

Once the saving process is completed, the Sandbox is not destroyed. The Sandboxing Middleware will reload the Sandbox Viewer so as to allow the user to complete the installation or save the remaining tools. It will also be possible to modify the just saved Tool to create, if needed, different and independent versions of it.

12. In case many tools are being provisioned at the same time, the user repeats the installation and saving procedure (steps 4 to 11) for each other Operating System in the Sandbox (i.e., for each tool).

13. After having saved all the tools, the user can destroy the Sandbox by clicking on "Destroy".

> **Warning:** Destroying the sandbox without having saved all the tools will result in unrecoverable loss of unsaved installed data.

14. The saved tools are now available in the Sandboxing Middleware Tools Repository.

> **Note:** Do not forget to update the "Available in Sandbox" flag in the platform catalogues (portal) to indicate that your Tools (or Model) is now available in the Sandboxing Middleware too.

6.2 Models Provisioning

In addition to provisioning a *Tool*, a Provider can also add a model to the Models Repository. A Model is an archive (.tar.gz or .zip) containing files (e.g., project files, tutorials, manuals ...) and folders.

To add a new model, the provider accesses the Models Repository from the home page and clicks on "Upload New Model" (*Figure* 38):

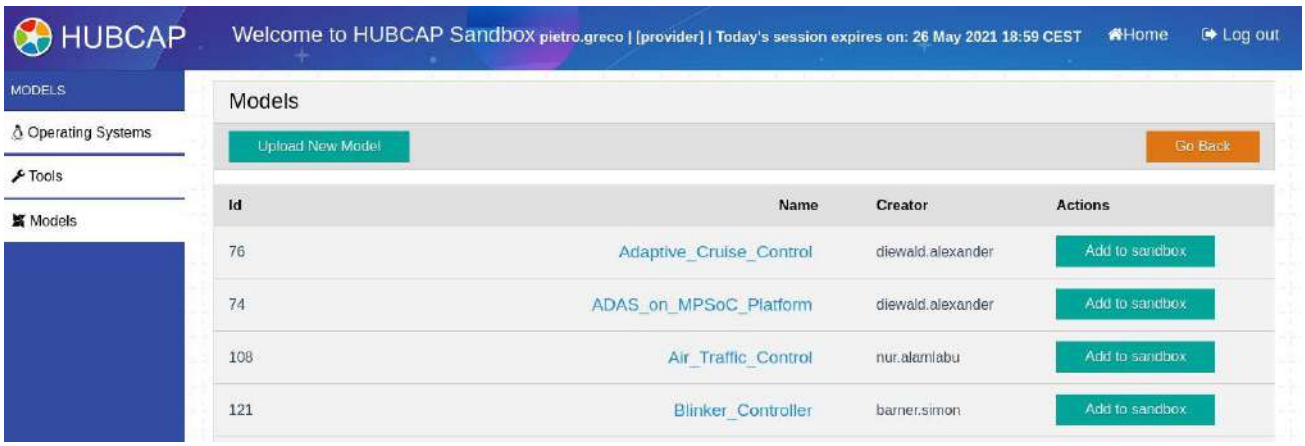


Figure 38- *Models repository*

then he is prompted with the following form (Figure 39):

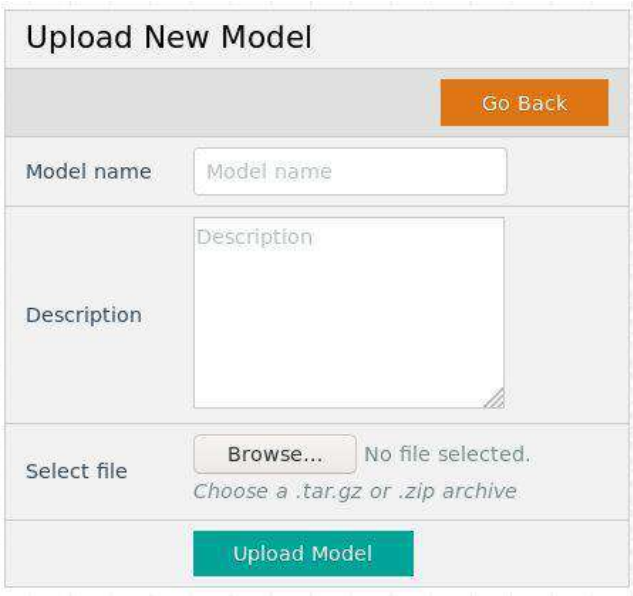


Figure 39 - Upload New Model form

He enters model name, description, selects the archive from his local computer and finally clicks on “Upload Model”. The model is now available in the Models Repository and accessible to the other users.

6.3 Repository Management

6.3.1 Destroying Items

From a repository, the user clicks on the item name to access the details page, then clicks on “Destroy” to destroy the item. He can only destroy those tools and models he has previously provided.

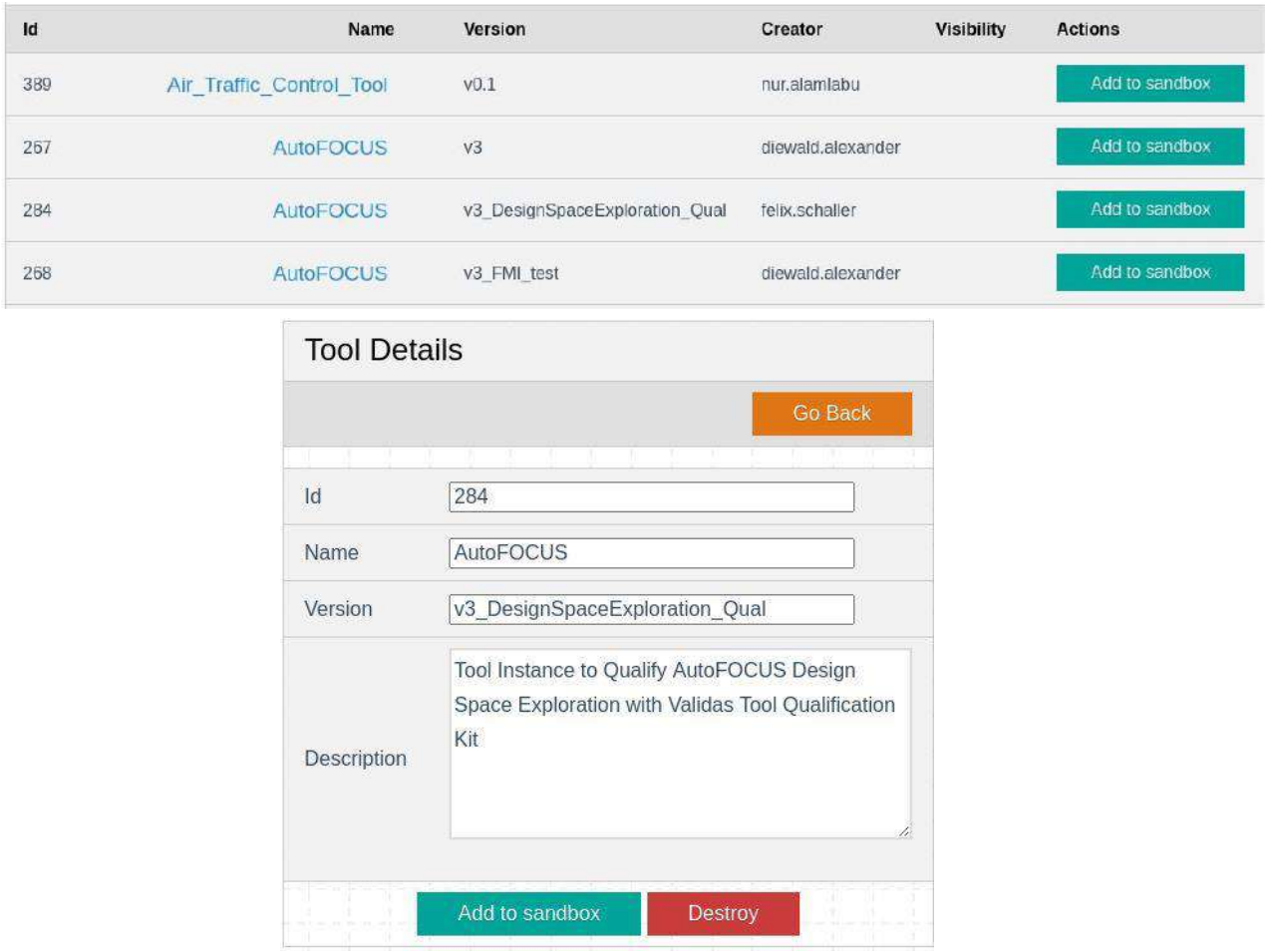


Figure 40 - Tools and Tool Details pages

6.3.2 Changing tools visibility

A Provider can change the visibility settings of their tools from the Tools repository (tools list) by clicking on the desired state: “Private” or “Public" (Figure 41):

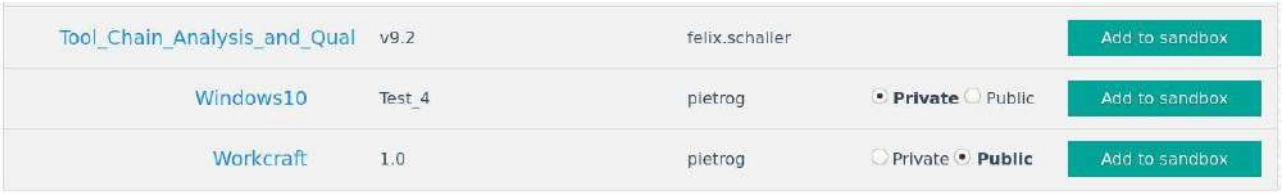


Figure 41 - Tools Repository: list of tools.