



Introduction à l'utilisation de la VM (VM)



Contenu

1 Objectifs	2
2 Introduction	2
3 Installation de Virtual Box	3
3.1 Download	3
3.2 Installation	4
3.3 Utilisation	5
4 Importation d'une machine virtuelle	6
4.1 Importation d'un fichier OVA	6
4.2 Configuration d'une machine virtuelle	7
5 Utilisation de la machine virtuelle DiD	9
5.1 Login	9
5.2 Programmes disponibles	9
5.2.1 Programmes principaux	9
5.2.2 Programmes facultatifs	9
5.3 Se connecter avec le VPN	9
5.4 Accès aux lecteurs du réseau des HEI	10
5.5 Utilisation du laboratoire de l'DiD	11
A GIT Commandes	12
AA Examen des modification et création d'une opération de validation	12
AB Synchronisation des changements	12
B Commandes Git les plus utilisées	13
BA Start a working area	13
BB Work on the current change	13
BC Examine the history and state	13
BD Grow, mark and tweak your common history	13
BE Collaborate	13
Bibliographie	14



1 | Objectifs

Ce document a pour but de montrer ce qu'est exactement une machine virtuelle, comment installer VirtualBox et comment importer et utiliser la machine virtuelle pour les laboratoires ELN.

2 | Introduction

En informatique, une machine virtuelle (VM) est l'encapsulation technique d'un système informatique dans un système informatique exécutable. La machine virtuelle simule l'architecture informatique d'un ordinateur réel ou hypothétique existant dans le matériel.

La couche d'abstraction entre l'ordinateur réel ou hôte sur lequel la machine virtuelle est exécutée et la machine virtuelle est appelée hyperviseur ou moniteur de machine virtuelle et sa mise en œuvre est purement matérielle, purement logicielle ou une combinaison des deux. L'hyperviseur permet généralement à plusieurs machines virtuelles de fonctionner simultanément sur une machine physique.

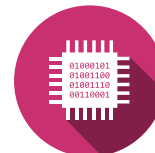
Contrairement aux émulateurs, les machines virtuelles fonctionnent directement sur l'unité centrale de l'ordinateur hôte et utilisent généralement les capacités de virtualisation de l'unité centrale. Avec les émulateurs, l'exécution est réalisée purement comme un logiciel, ce qui signifie qu'une architecture informatique différente de celle de l'ordinateur hôte peut également être émulée.



En bref, une machine virtuelle est un ordinateur complet fonctionnant dans une fenêtre de l'ordinateur hôte

Pour notre cours, nous avons besoin du programme [VirtualBox](#) de [Oracle](#). Il existe d'autres fabricants sur le marché qui proposent des solutions pour les machines virtuelles. Voici une liste incomplète:

- [VMWare](#)
- [Parallels](#)
- [Windows WSL](#)
- [Docker](#)



3 | Installation de Virtual Box

Tout d'abord, le programme de l'hyperviseur doit être installé sur l'ordinateur hôte (votre ordinateur).

3.1 Download

Téléchargez la dernière version de VirtualBox sur leur site web <https://www.virtualbox.org/>.



Fig. 1. – VirtualBox page d'accueil

- Sélectionnez le fichier en fonction de votre système d'exploitation



Fig. 2. – Site de téléchargement de VirtualBox

- Enregistrez le fichier en local sur l'ordinateur

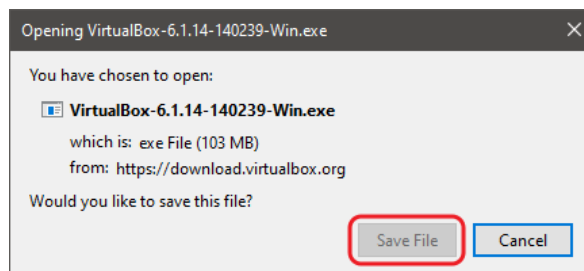


Fig. 3. – Dialogue pour la sauvegarde du fichier

3.2 Installation

Lancez le fichier que vous venez de télécharger **VirtualBox-<Version>-<Betriebssystem>.exe**
P.ex. **VirtualBox-6.1.15-140239-Win.exe**

- Appuyez 3 fois sur **Next**

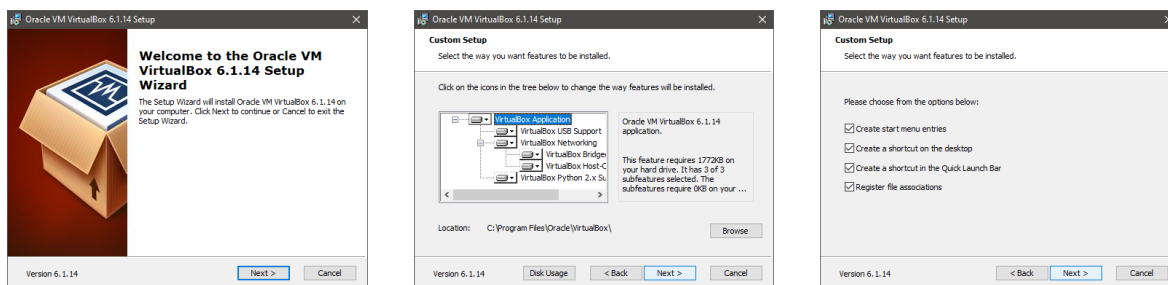


Fig. 4. – VirtualBox étapes d'installation 1-3

- Une interface réseau est alors automatiquement installée. Cela est nécessaire pour permettre à la machine virtuelle d'accéder au réseau de l'ordinateur hôte. Appuyez ici sur **Yes**.
- Après cela, le programme d'installation a été configuré et vous pouvez appuyer sur **Install**

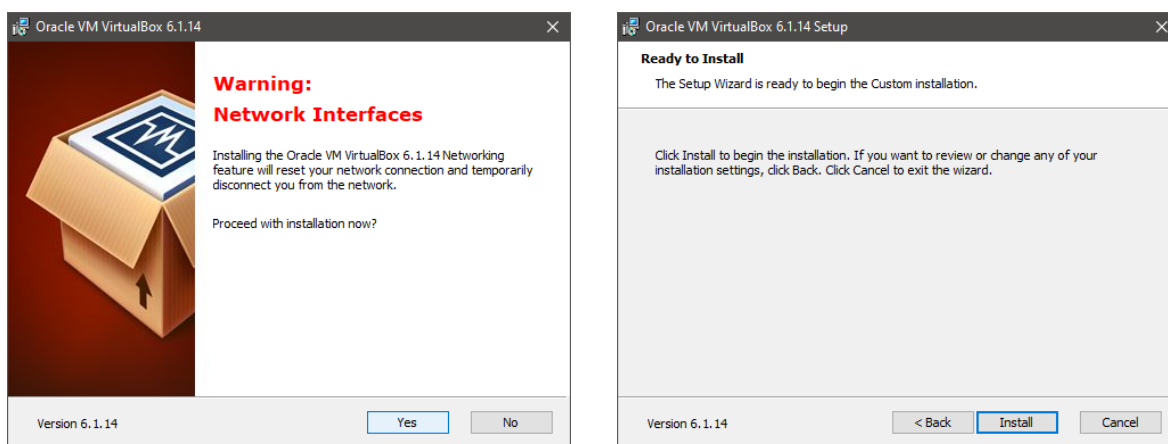
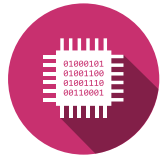


Fig. 5. – VirtualBox étape d'installation 4-5



Vous venez d'installer avec succès VirtualBox

3.3 Utilisation

Le programme peut être lancé via l'icône du bureau.



Fig. 6. – VirtualBox Icon

La fenêtre principale vous permet de créer, d'importer, de supprimer et de démarrer des machines virtuelles.

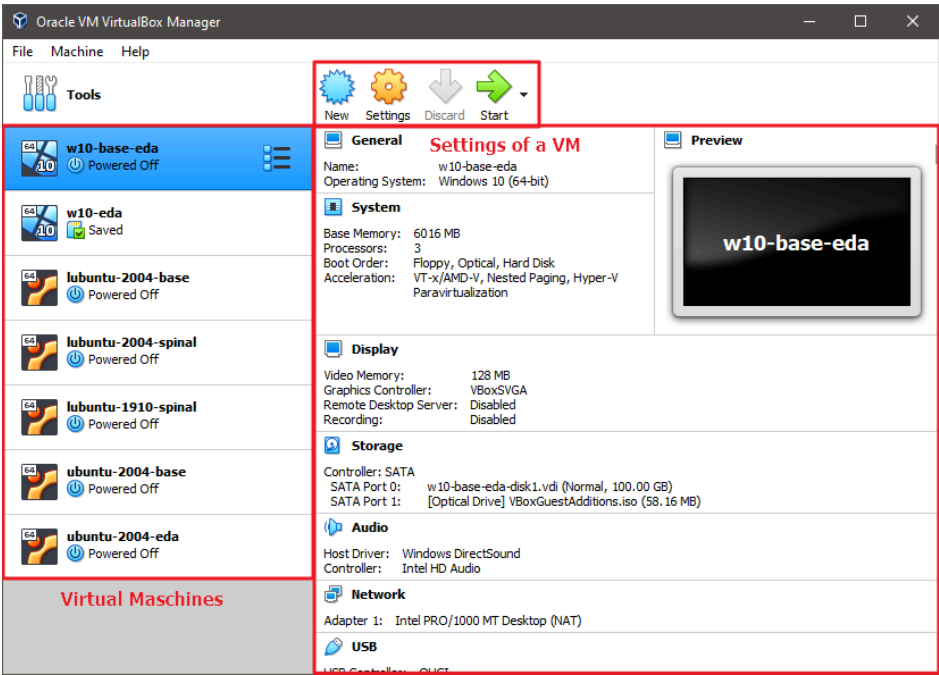
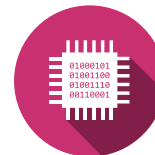


Fig. 7. – VirtualBox GUI



4 | Importation d'une machine virtuelle

4.1 Importation d'un fichier OVA

Le format OVF (Open Virtualization Format) est une norme ouverte pour le conditionnement et la distribution d'appareils virtuels ou de logiciels généraux fonctionnant dans des machines virtuelles. Cette norme a été élaborée par la Distributed Management Task Force (DMTF).

La norme décrit un « format ouvert, sécurisé, portable, efficace et extensible pour le conditionnement et la distribution de logiciels fonctionnant dans des machines virtuelles. La norme OVF n'est pas limitée à des hyperviseurs ou des architectures de processeur spécifiques. L'unité qui intervient dans le conditionnement et la distribution est appelée paquet OVF. Un paquet OVF peut contenir un ou plusieurs systèmes virtuels, chacun d'eux pouvant être déployé dans une machine virtuelle.

- Pour importer le fichier OVA, appuyez sur **File** et **Import Appliance**

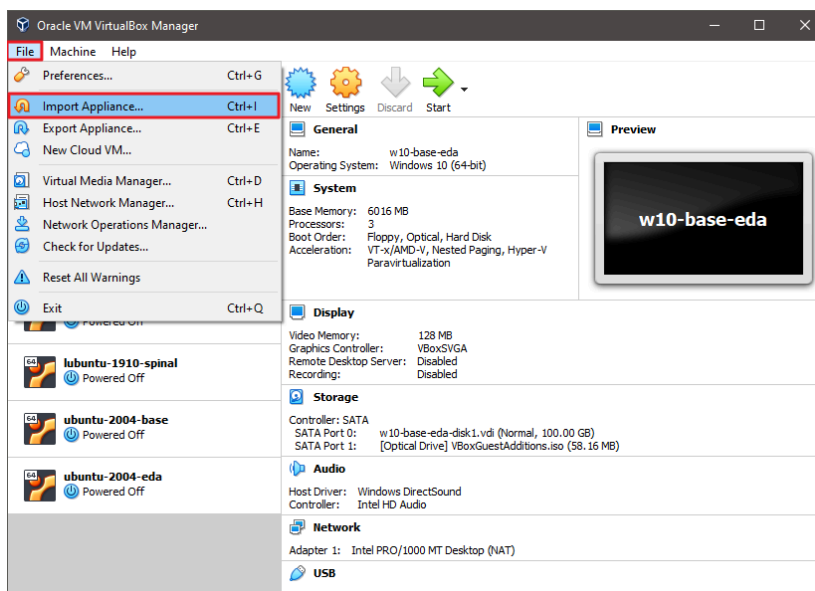


Fig. 8. – Import Appliance

- Trouvez le fichier *.ova qui vous a été envoyé
- Sélectionnez l'emplacement des machines virtuelles et appuyez sur **Import**



Notez que la machine virtuelle peut occuper beaucoup de mémoire. Pour l'image standard du laboratoire DiD, il faut au moins 26 Go de mémoire.

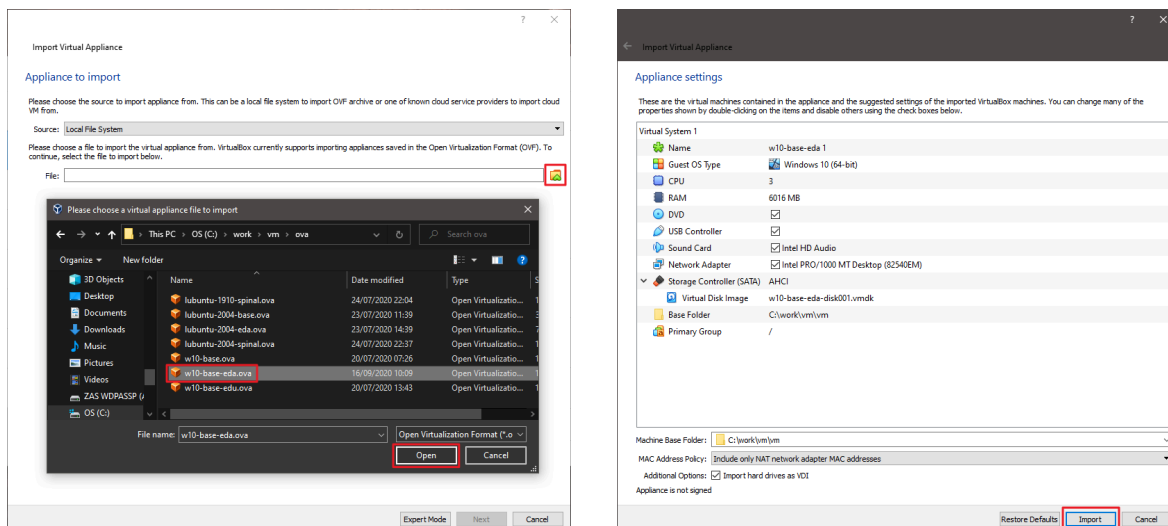
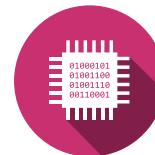


Fig. 9. – Importation d'un fichier OVA existant



Vous venez d'importer avec succès une machine virtuelle

4.2 Configuration d'une machine virtuelle

Une machine virtuelle utilise les ressources (CPU, RAM, espace disque) de l'ordinateur hôte. Plus vous fournissez de ressources à la machine virtuelle, moins l'ordinateur hôte en a. Choisissez une configuration de cœurs de processeurs et d'espaces de mémoire vive (RAM) en fonction de votre ordinateur.

- Ouvrez les paramètres de votre machine virtuelle

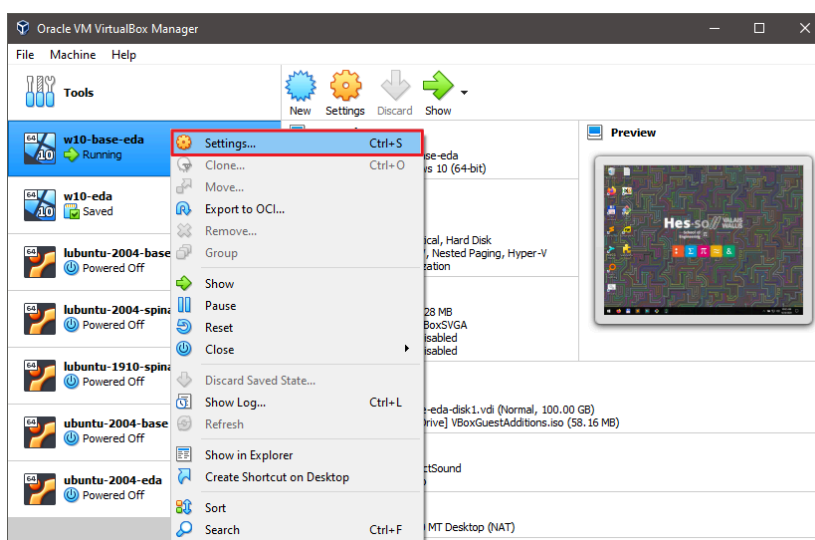


Fig. 10. – Configuration d'une machine virtuelle

Une configuration recommandée est :

- ≥ 2 cœur CPU



- ≥ 4 GB RAM

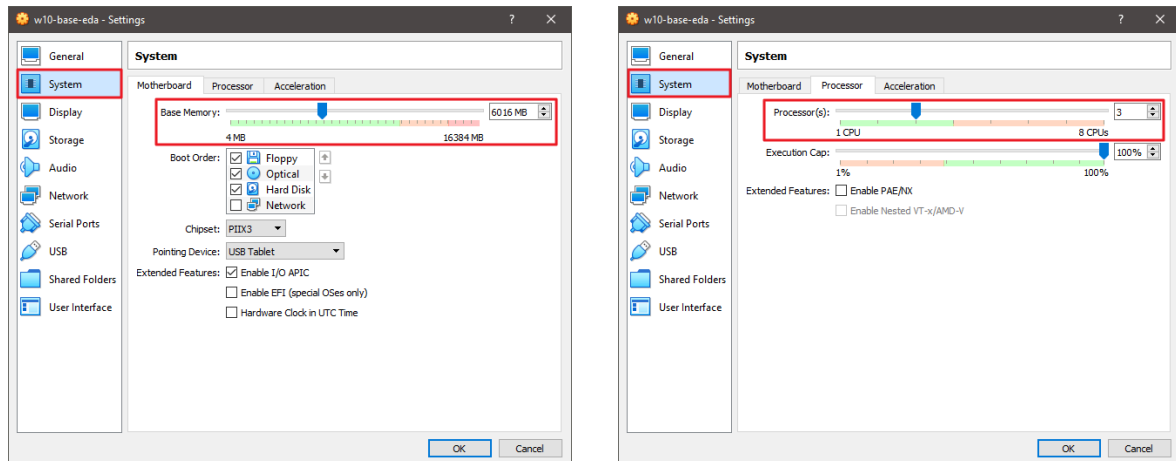
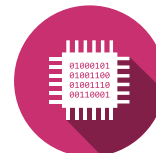


Fig. 11. – Configuration des cœurs de l'unité centrale et de la mémoire RAM



5 Utilisation de la machine virtuelle DiD

5.1 Login

L'utilisateur s'appelle **uadmin** et possède le mot de passe **e!n**.





L'utilisateur a des droits d'administrateur.
« With great power comes great responsibility »











5.2 Programmes disponibles

L'image fournit de nombreux programmes différents, qui ne sont pas tous utilisés pour le laboratoire.

5.2.1 Programmes principaux

-  Mentor HDL-Designer - Programme de conception pour l'électronique numérique
-  Mentor Modelsim - Programme de simulation pour l'électronique numérique

5.2.2 Programmes facultatifs

-  HESO VPN Software pour établir une connexion VPN avec l'HEI
-  Mozilla Firefox - navigateur web
-  Hex Viewer et Editor
-  Navigateur de fichiers
-  Logiciel permettant de localiser rapidement les fichiers sur le disque dur
-  Editeur de texte
-  Outil graphique GIT
-  PDF Viewer
-  Logiciel permettant d'analyser où l'espace mémoire est utilisé
-  Ligne de commande pour les commandes de git

5.3 Se connecter avec le VPN

- Démarrer  PulseSecure



- Se connecter à l'AAI Login <prenom>.<nom>



Le login AAI se compose des 8 premiers caractères de votre prénom et nom.

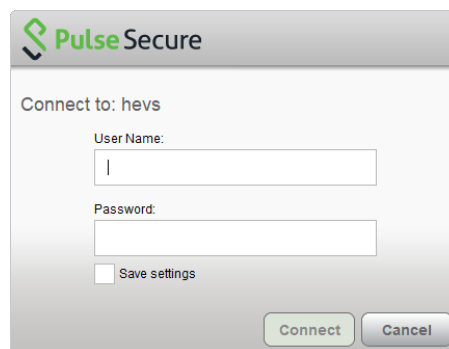
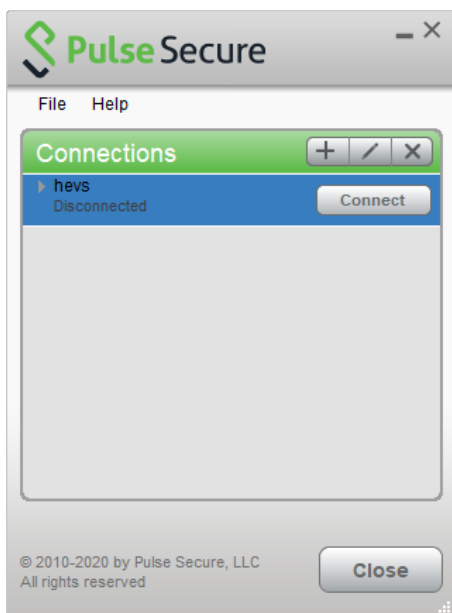


Fig. 12. – Connexion VPN Pulse Secure

5.4 Accès aux lecteurs du réseau des HEI

- Si vous n'êtes pas dans le réseau de l'école, vous devez d'abord établir la connexion VPN avec

Pulse Secure  voir [Chapitre 5.3](#)

- Ouvrez un explorateur de fichiers et cliquez sur le lecteur souhaité. Connectez-vous ensuite avec le login AAI <prenom>.<nom>

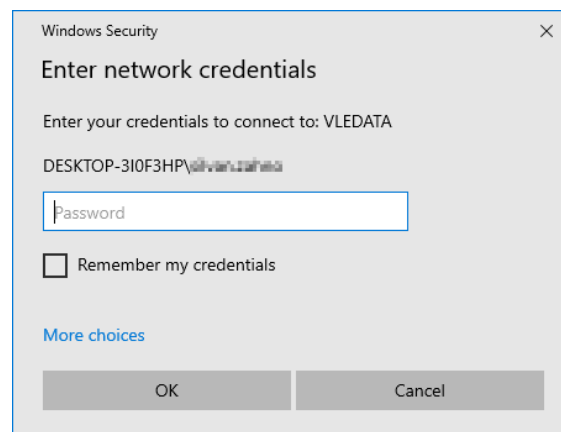
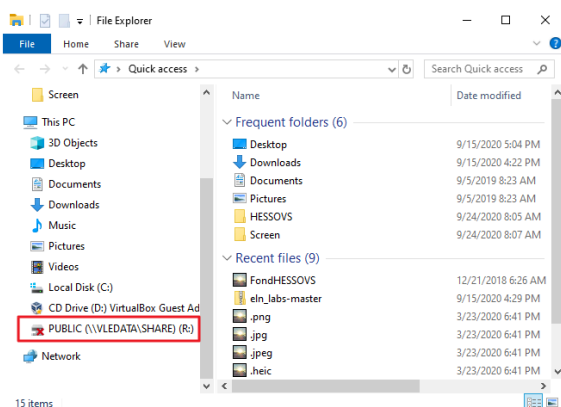


Fig. 13. – Accès aux lecteurs du réseau des HEI



5.5 Utilisation du laboratoire de l'DiD

Une copie des fichiers DiD Labor se trouve déjà dans le dossier **C:\work\DiD_labs**.

Le lien pour démarrer le programme se trouve également sur le bureau **C:\Users\uadmin\Desktop\DiD_Labs.lnk**.



Fig. 14. – Icône pour lancer DiD Labs

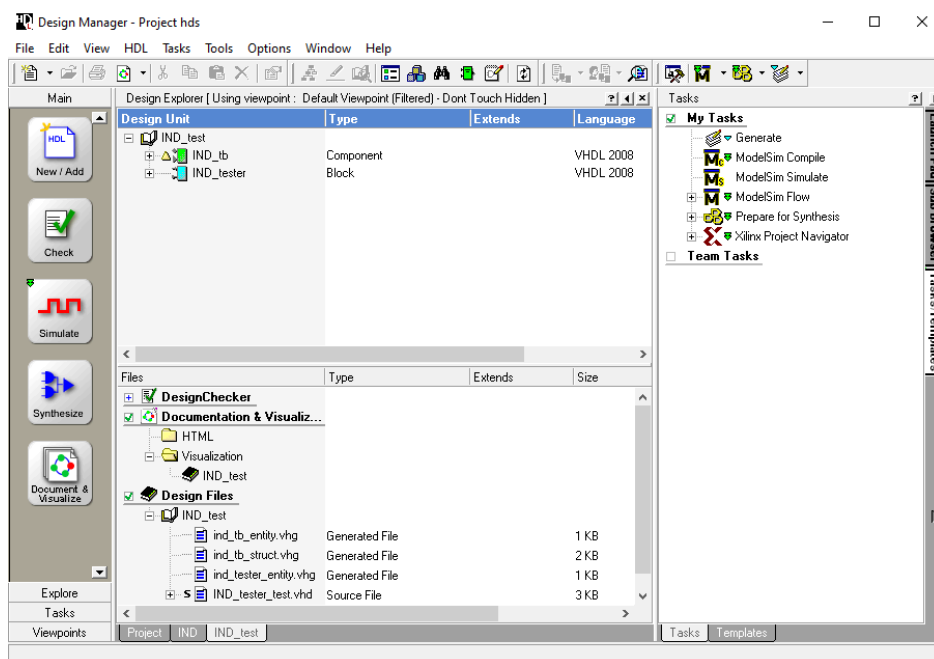


Fig. 15. – HDL-Designer DiD Labs



A | GIT Commandes

[Github git cheatsheet \[1\], \[2\]](#)

AA Examen des modification et création d'une opération de validation

```
git status
```

Liste tous les fichiers nouveaux ou modifiés qui sont prêts à être commit.

```
git diff
```

Affiche les modifications de fichiers qui n'ont pas encore été indexées.

```
git add [file]
```

Ajoute le fichier au versionning.

```
git diff --staged
```

Affiche les différences entre l'index (« staging area ») et la version actuelle du fichier.

```
git reset [file]
```

Retire le fichier de l'index (« staging area ») mais ne le supprime pas du disque.

```
git commit -m "[descriptive message]"
```

Inclut tous les fichiers actuellement indexés de façon permanente dans l'historique des versions.

AB Synchronisation des changements

Enregistrement d'un référentiel externe (URL) et échange de l'historique du repository.

```
git fetch [remote]
```

Télécharge l'historique complet d'un repository externe.

```
git merge [remote]/[branch]
```

Intègre la branche externe dans la branche locale.

```
git push [remote] [branch]
```

Pousse la branch locale (donc tous les commits de celle-ci) sur GitHub.

```
git pull
```



Récupération de l'historique du repository externe et intégration des modifications sur le repository local.

B | Commandes Git les plus utilisées

BA Start a working area

- **clone** - Clone a repository into a new directory
- **init** - Create an empty Git repository or reinitialize an existing one

BB Work on the current change

- **add** - Add file contents to the index
- **mv** - Move or rename a file, a directory, or a symlink
- **reset** - Reset current HEAD to the specified state
- **rm** - Remove files from the working tree and from the index

BC Examine the history and state

- **log** - Show commit logs
- **show** - Show various types of objects
- **status** - Show the working tree status

BD Grow, mark and tweak your common history

- **branch** - List, create, or delete branches
- **checkout** - Switch branches or restore working tree files
- **commit** - Record changes to the repository
- **diff** - Show changes between commits, commit and working tree, etc
- **merge** - Join two or more development histories together
- **rebase** - Reapply commits on top of another base tip
- **tag** - Create, list, delete or verify a tag object signed with GPG

BE Collaborate

- **fetch** - Download objects and refs from another repository
- **pull** - Fetch from and integrate with another repository or a local branch
- **push** - Update remote refs along with associated objects



Bibliographie

- [1] gitlab, « Git Cheatsheet ». 2023.
- [2] « GitHub Git Spickzettel ». Consulté le: 25 avril 2023. [En ligne]. Disponible sur: <https://training.github.com/downloads/de/github-git-cheat-sheet/>