

# Information, Calcul et Communication

## Composante Pratique: Programmation C++

Faculté Informatique et Communications

Groupe de recherche en Interaction Immersive

EPFL-IIG

**Dr Ronan Boulic**

**RLC** vendredi 8h-9h **SAUF** les dates suivantes:

<b>15.10.21:</b> cours donné au	<b>STCC plénière Garden</b>
<b>12.11.21:</b> cours donné au	<b>STCC Auditoire A</b>
<b>19.11.21:</b> cours donné au	<b>STCC Auditoire A</b>
<b>24.12.21:</b> cours donné au	<b>SG 1</b>

## Plan

- **Rappel**
  - Organisation : Théorie et Pratique avec sites moodle
- **But de la composante Pratique**
  - Pourquoi C++ ?
- **Exploitation du MOOC**
  - Documents de référence MOOC
  - Comment travailler en classe inversée
  - Le cours ne se limite pas au MOOC
- **La machine virtuelle (VM)**
- **Le Mini-Projet**

# Organisation du cours CS-119(c) selon ses deux composantes :

	Théorie	Pratique (C++)
Moodle	<a href="#">Moodle: 14002</a>	<a href="#">Moodle: 6831</a>
Complément	Livre de référence « <a href="#">Découvrir le Numérique</a> »	<a href="#">MOOC</a> : inscription dès la semaine 1 Utilisation entre les semaines 2 et 11
Infos organisation	<a href="#">News moodle</a>	<a href="#">News moodle</a>
Cours live / zoom	<a href="#">Webinar theorie</a> <b>mardi 11-13h</b> Q&A avec 2 doctorants	<a href="#">Webinar pratique</a> <b>vendredi 8-9h</b> Q&A avec 2 doctorants
QCM pendant cours	Site web <a href="#">speakup.info</a> => lien Try-it => <a href="#">SpeakUp : Room 21158</a>	
<b>Questions et Réponses (Q&amp;A)</b>	<u>Pendant les heures de cours</u> : utiliser le Q&A du webinar <u>Le reste du temps, y compris exos, TP, support</u> : <a href="#">Discourse</a>	
Exo & TP présentsiels	<a href="#">Liste des salles</a>	
Exo & TP en-ligne	Meeting <a href="#">Zoom</a> pour les personnes qui ne sont pas autorisées en présentiel	
Support en semaine	Jeudi 9-10h et vendredi 9-10h dès le 30 Septembre	

# Buts de la composante pratique

- **Fondements de la Programmation en C++**
  - Maîtriser un langage de programmation et son environnement de développement
  - Conception (pseudocode) et mise au point d'algorithmes (codage et tests)
  - Mise en oeuvre d'un mini-projet (sem 7-14)
- **Savoir utiliser un ordinateur sous LINUX**
  - **Machine Virtuelle (VM)** à utiliser sur votre propre ordinateur/laptop, même s'il fonctionne sous Windows ou OS X.
- **Illustration des thèmes de ICC-Théorie**
- **Acquérir une bonne méthodologie de travail**
  - régularité, équilibre entre matières, rigueur

# Pourquoi C++ ?

- C++ prend le relai du langage C (langage procédural très utilisé pour les systèmes embarqués mais manquant d'éléments pour structurer le développement de code fiable)
  - Beaucoup de points communs entre C et C++ : syntaxe / orienté performances temps-réel
- C++ permet de comprendre ce qui se passe en *mémoire centrale* pendant l'exécution du programme
- C++ est de haut-niveau ( Programmation Orientée Objet abordée au second semestre)
- Comparé à Python (très en vogue pour mettre en oeuvre le machine learning), C++ offre de meilleures performances *temps-réel*



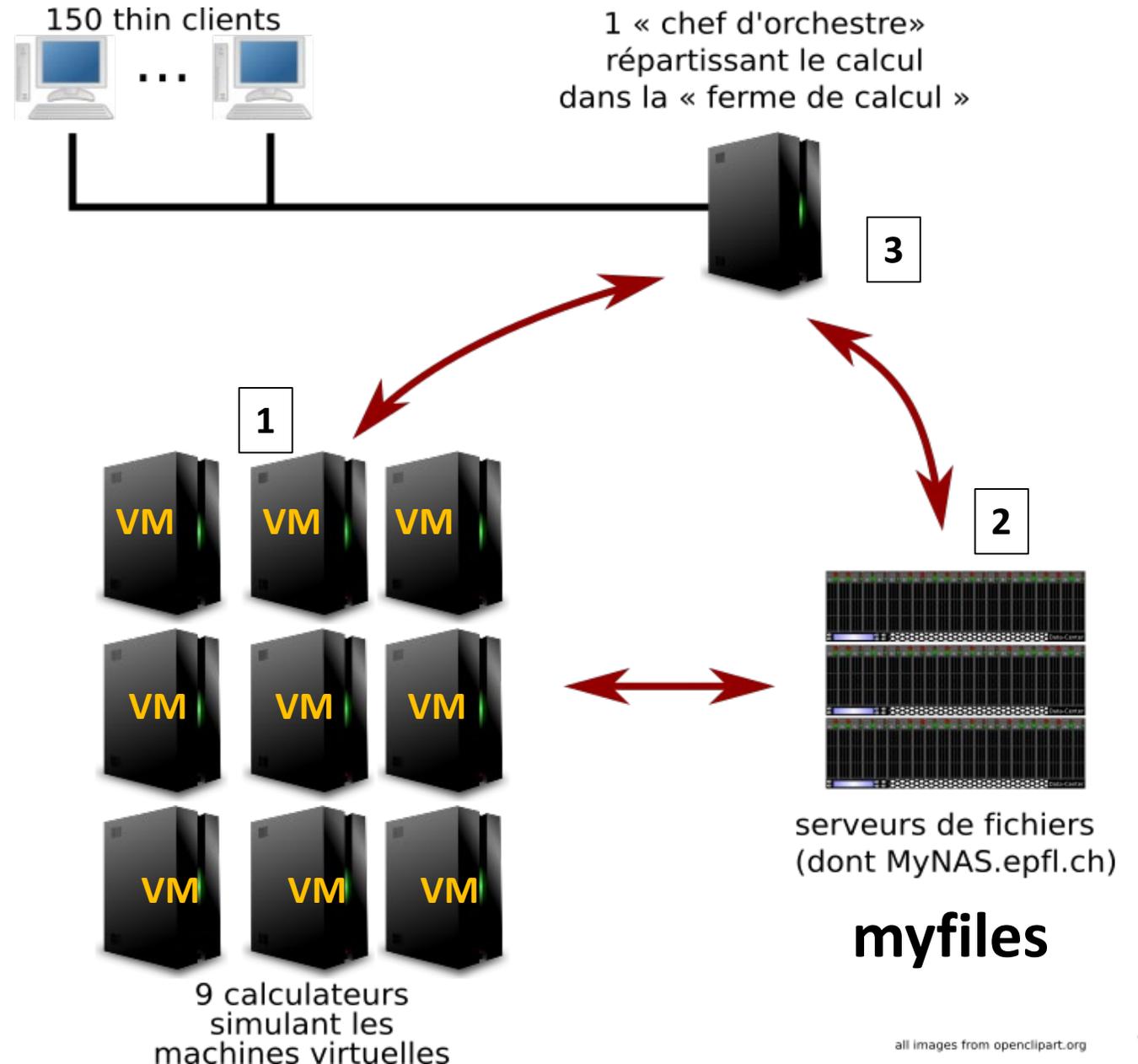
# Méthode de Travail : classe inversée en phase MOOC

- Chaque semaine, selon le planning :
  - **AVANT le cours du vendredi matin:**
    - voir les vidéos de la leçon MOOC, faire les quizzes (dans et hors vidéo)
      - Entre 1h30 et 2h30 de travail
    - commencer les exercices
      - Entre 30 min et 1h30
  - **Vendredi 8h15-9h00** : classe inversée = rappel, complément et codage sur la VM
    - Poser des questions, participer à la résolution d'exercice
  - **Vendredi 15h15-17h00**: séance d'exercice avec assistants
    - Finir les exercices recommandés si possible.
  - **AVANT le lundi**
    - Finir les exercices recommandés
    - Faire le devoir MOOC s'il y en a un (optionnel mais recommandé)

- Sujets et exercices complémentaires pour l'après MOOC
  - Approfondissement technique sur ce qui se passe en mémoire centrale pendant l'exécution du programme
  
- Autres ressources:
  - Fiches, conseils recherche de bug, conventions (style du code)
  - Code source utilisé en classe inversée (live coding sur la VM)

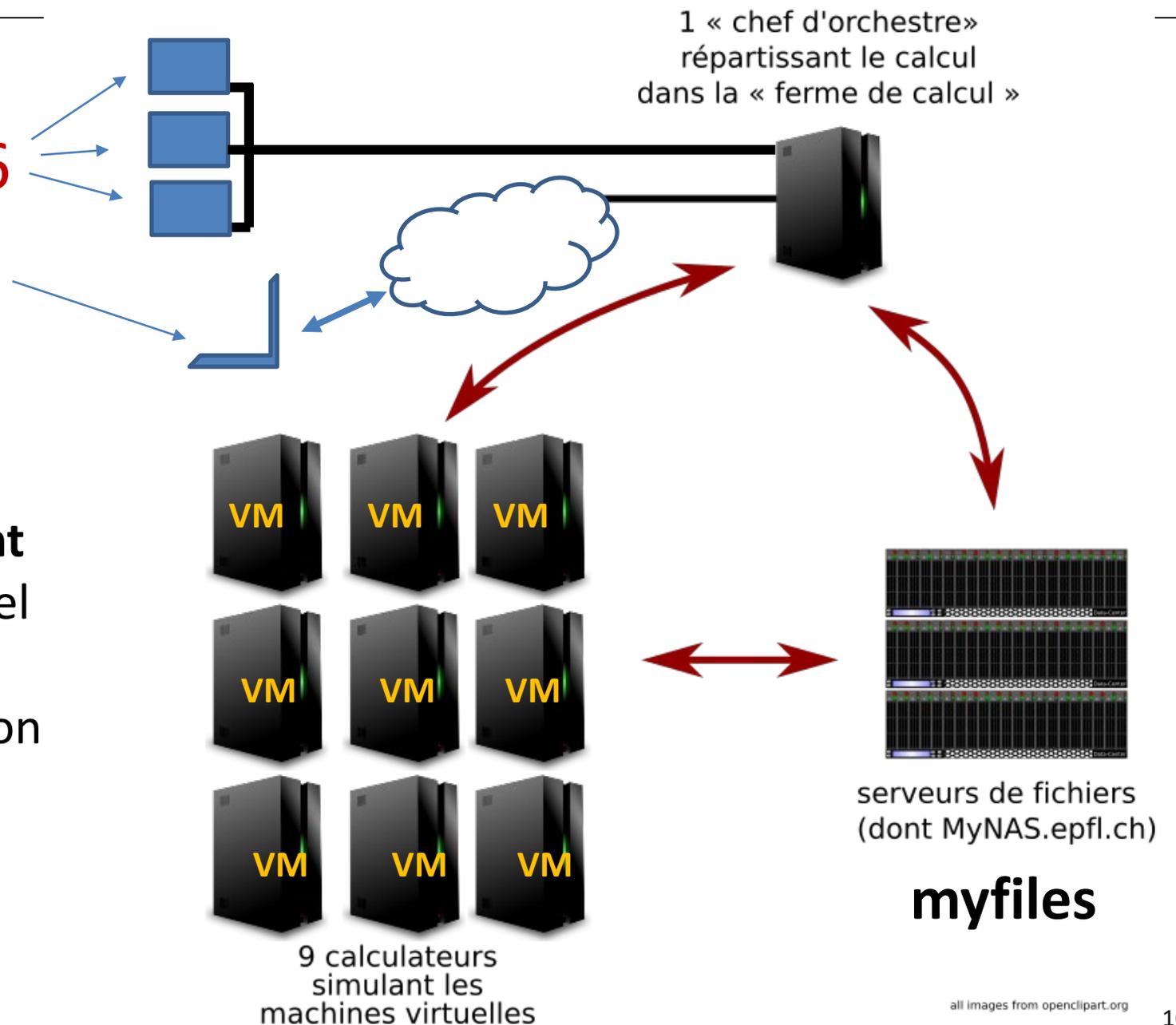
# La Machine Virtuelle (VM)

- **Utilisateur** : Interface (thin client)
  1. **Calcul** pour les machines virtuelles (VM) temporaires selon les besoins
  2. Serveurs de **stockage** permanent des fichiers des utilisateurs (myNAS visible sous **myfiles**)
  3. Serveur *chef d'orchestre* responsable de **répartir les charges** de calcul de VM entre les calculateurs, et de faire le **relai** entre les utilisateurs, leur VM et leur compte myNAS.



# Accès à une VM depuis les salles **CO4-5-6** ou son **laptop personnel**

- Navigateur + **vdi.epfl.ch**
- + Application **Horizon thin client** installée sur un laptop personnel
- Solution rapide à installer sur son laptop (série1 TP)



# Mini-projet «SAME GRANMA»

- Travail INDIVIDUEL entre les semaines 7 et 14
  - les fonctions et les types **vector** et **string** suffiront
  - rendu téléchargé sur moodle (deadline 24/12)
  - utilisation du détecteur de plagiat
- Conception à l'aide du pseudocode d'un algorithme (fourni)
- Structuration du programme à l'aide de deux grands principes
  - Abstraction et Ré-utilisation
- Méthode de mise au point de projet
  - Tests systématiques
- Evaluation à l'aide d'un autograder que vous pouvez aussi utiliser:
  - Execution des fichiers tests publics et non-publics
- Evaluation du style du code par les assistants
  - Code: lisibilité, style, conventions du cours

# Correspondance C++ [MOOC, Projet]

Mini-Projet	week	MOOC	Cours Pratique Vendredi 8h-9h	TP Vendredi 15h-17h
	1	<i>Inscription au MOOC</i>	Intro générale	Série1: Linux et VM
	2	Sem1: variables	Var / expr / mémoire	Série2: Variable, expr
	3	Sem2: if	If , switch, ++i	Série3: If , switch, ++i
	4	Sem3: for - while	For, while	Série4: For, while
	5	Sem4: fonctions()	Fonctions (1)	Série5: Fonctions (1)
Redirection (vue en TP)	6		Fonctions (2)	Série6: Fonctions (2)
Donnée & Démo	7	Sem5: vector, array	vector	<b>EXAMEN ECRIT</b> <b>5 novembre 14-17h</b>
	8	Sem6: string, struct	Array, string	Série7: vector <i>(décalé de 16h à 18h)</i>
	9		Structure	Série8: string
	10	Sem7: pointeur	Pointeur(1)	Série9: structure
	11		Pointeur(2)	Série10: pointeur
	12		Entrée-sortie	Série11: Entrée-sortie <i>(décalé de 16h à 18h)</i>
	13		Révisions	<b>EXAMEN ECRIT</b> <b>17 décembre 15-18h</b>
<b>Rendu 24/12 midi</b>	14		Analyse des performances de vector	<i>vacances</i>