

Document de conception détaillée

Synthlab

Chevallier Raphaël
Chevalot Emilie
Guerra Elia
Hardoüin Clément
Kwete Dany
Le Hô Romain

Sommaire

[Description du projet](#)

[Principes généraux](#)

[Description du logiciel](#)

[Fonctionnalités](#)

[Technologies employées](#)

[Liste des composant logiciels](#)

[Architecture](#)

[Modèle métier](#)

[Adaptation de la bibliothèque JSyn](#)

[Modèle d'interaction PAC](#)

[Abstraction](#)

[Architecture PAC d'un module](#)

[Scénario d'interaction PAC](#)

[Conclusion](#)

Préface

But

Le document de conception détaillée fournit la structure interne des composants logiciels utilisés pour mettre en oeuvre le logiciel.

L'architecture générale, les outils, les patrons de conception et la description des différents modules y sont présentés.

Audience

Ce document s'adresse au personnel et étudiants de l'ISTIC

Description du projet

Le projet “Synthlab” consiste à créer un logiciel capable de synthétiser des flux audio en utilisant un ensemble de modules de traitement audio. Ce logiciel vise à simuler une architecture matérielle analogique connue sous le nom de synthétiseur analogique à synthèse soustractive.

Principes généraux

L'utilisateur de l'application dispose d'une bibliothèque de modules de traitement.

Chaque module dispose de ports d'entrées, de ports de sorties et de paramètres de réglage.

L'utilisateur peut assembler différents modules de façon à construire une machine de synthèse en créant des connexions entre les ports de sortie et les ports d'entrée.

Les connexions font circuler des flux audio entre les modules. Certains modules sont des sources (émetteurs) et d'autre des puits (récepteurs).

Les modules doivent être paramétrables au moyen d'une interface graphique reprenant les codes des synthétiseurs analogiques physiques.

Description du logiciel

Fonctionnalités

Les fonctionnalités principales de l'application sont les suivantes :

- Ajout / déplacement / suppression de modules de traitement audio
- Réglage des modules par le biais d'une interface graphique
- Connexion des modules à l'aide de câbles
- Choix de la couleur des câbles
- Sauvegarde et chargement de montages
- Enregistrement d'un signal dans des fichiers audio

Technologies employées

L'application Synthlab est réalisée en **Java** et repose sur la bibliothèque de synthèse audio **JSyn** pour les différents modules de traitements audio.

Liste des composant logiciels

Synthlab se décompose en un ensemble de composants logiciels interconnectés.

Le **Synthétiseur** est le composant principal contenant tous les autres.

Les **Modules** sont les composants de traitement audio. Ils contiennent un ensemble de **Ports** d'entrée et de sortie permettant de recevoir et de transmettre des signaux sonore aux autres modules.

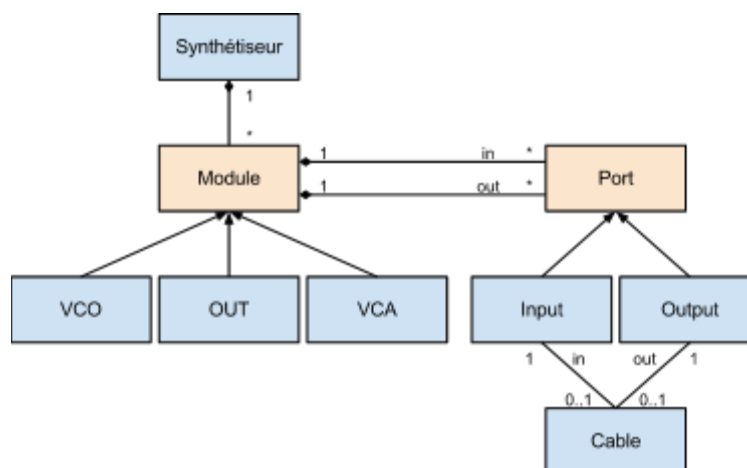
Les modules implémentés dans la version courante de synthlab sont les suivants :

- **VCO** : Générateur de signaux de différente forme.
- **OUT** : Module transmettant un signal vers la carte son de l'ordinateur.
- **REC** : Module permettant l'enregistrement d'un signal audio vers un fichier wav.
- **REP** : Module répliquant le signal d'entrée sur les ports de sortie.
- **VCA** : Module permettant de moduler l'amplitude du signal d'entrée.
- **EG** : Générateur d'enveloppe.
- **AudioScope** : Visualisateur de signal sonore.
- **VCF LP** : Filtre passe bas.
- **VCF HP** : Filtre passe haut.
- **MIX** : Mixeur permettant d'additionner plusieurs signaux.
- **NOISE** : Générateur de bruit blanc.

Afin de relier les modules entre eux, des **câbles** peuvent être reliés aux ports d'entrée et de sortie des modules.

Architecture

Modèle métier



Adaptation de la bibliothèque JSyn

Afin de simplifier le développement au niveau du domaine métier, la librairie JSyn, gratuite pour des projets open-source (licence GPL ou commercial), a été choisie.

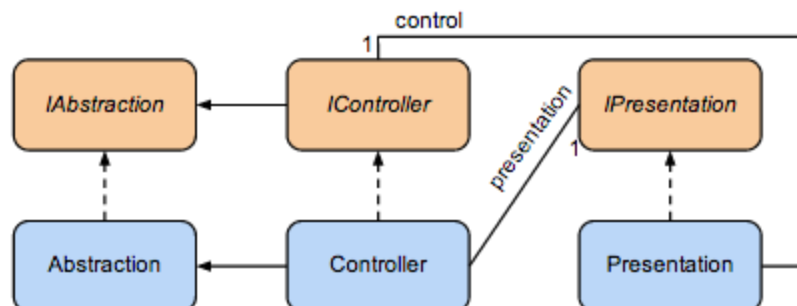
Cette librairie fournit toutes les fonctions de base permettant de construire un synthétiseur audio. Les interactions avec cette librairie se situent au niveau de l'abstraction dans le modèle PAC choisi. L'abstraction joue alors le rôle d'Adapter dans l'application afin de diminuer les dépendances avec cette librairie et ainsi avoir la possibilité d'en changer facilement.

Modèle d'interaction PAC

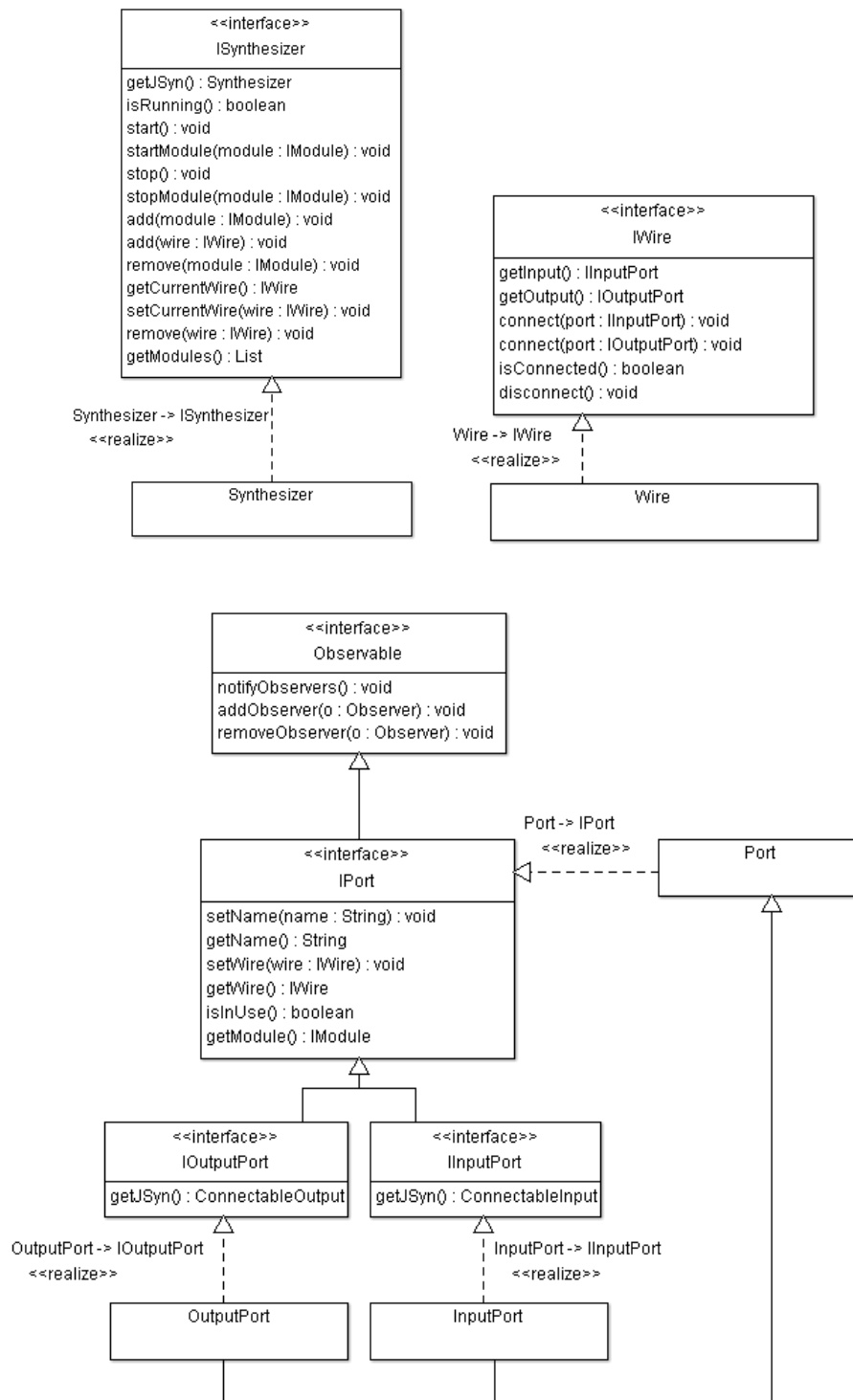
Afin de diminuer le couplage entre la couche métier et la couche de présentation de l'application, l'architecture PAC + Proxy + Héritage a été choisie.

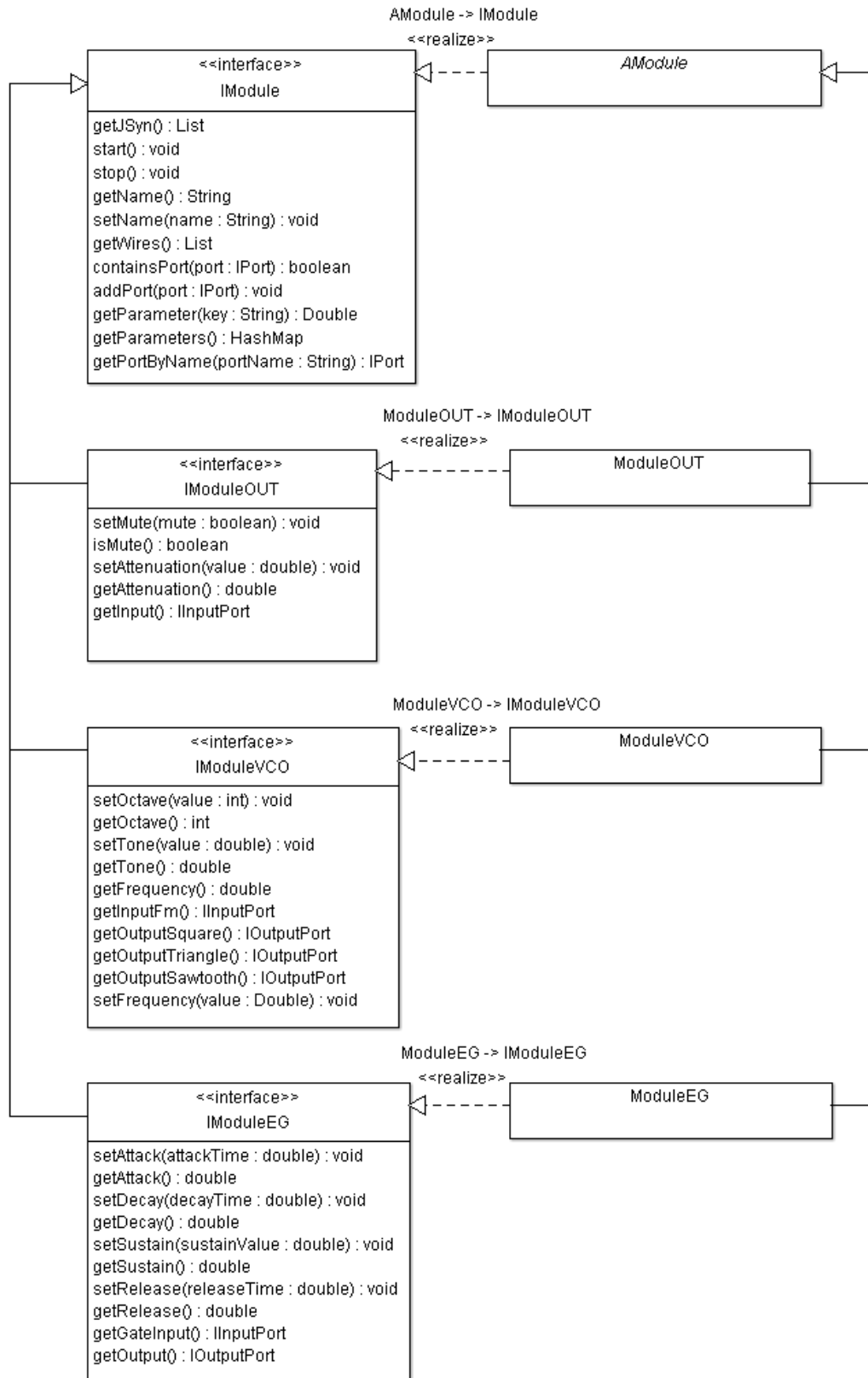
Chaque composant logiciel est donc découpé en trois éléments.

- L'abstraction correspond au coeur logique du composant
- Le contrôleur joue le rôle de médiateur entre la présentation et l'abstraction
- La présentation permet l'affichage du composant et informe le contrôleur des modifications apportées

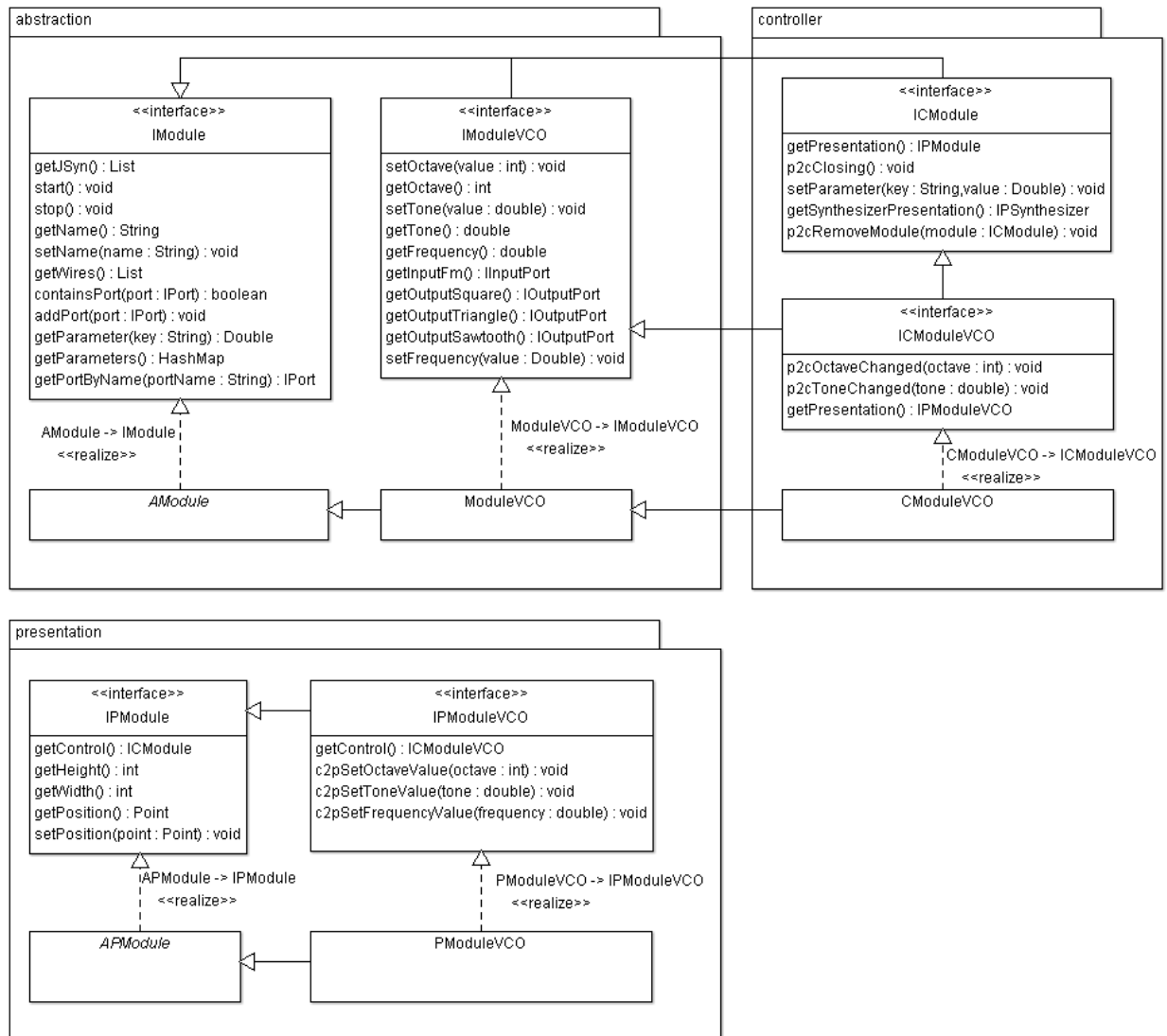


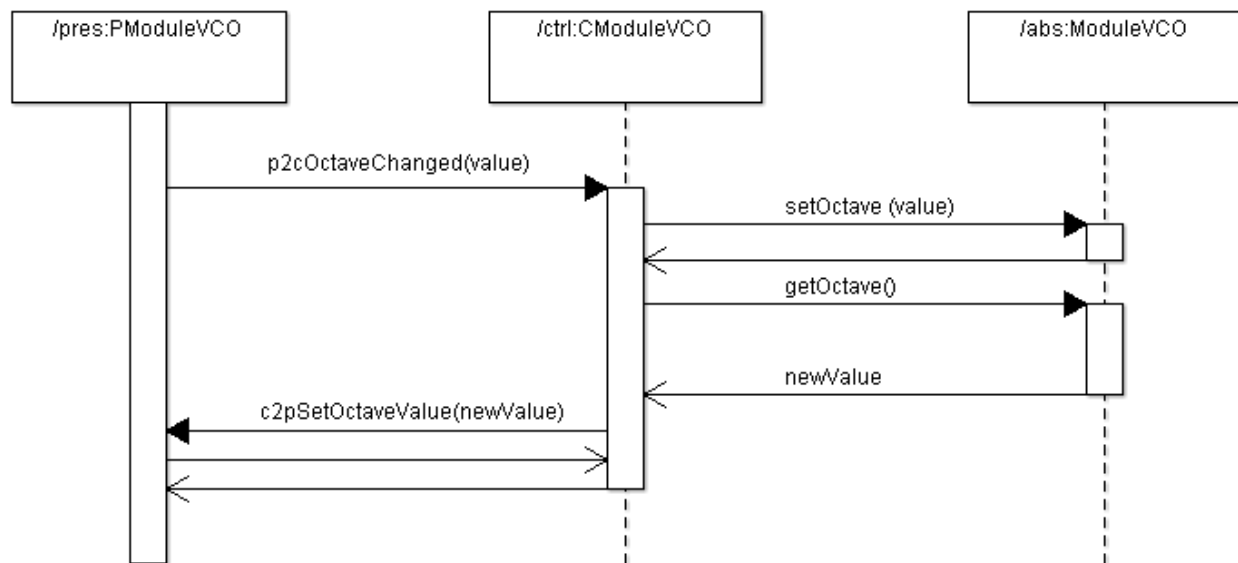
Abstraction





Architecture PAC d'un module



Scénario d'interaction PAC**Conclusion**

L'ensemble des modules utilise cette architecture. Cela permet une bonne séparation des préoccupations et diminue les couplages. Il est ainsi simple et rapide d'étendre l'application en ajoutant de nouveaux modules.