

# Project report: Coupling ScimBa and Feel++

Helya Amiri, Rayen Tlili

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Objectives . . . . .	2
1.2	Roadmap . . . . .	2
<b>2</b>	<b>Getting started</b>	<b>2</b>
2.1	Creating a Docker container . . . . .	2
2.2	Exploring Feel++ toolboxes . . . . .	3
<b>3</b>	<b>Bibliography</b>	<b>4</b>

# 1 Introduction

This report presents the objectives, approach, and roadmap for the coupling of ScimBa and Feel++ libraries. ScimBa is a project aimed at integrating machine learning techniques with traditional scientific computing methods, while Feel++ is a C++ implementation of Galerkin methods for solving partial differential equations (PDEs). The coupling of these two libraries is expected to enhance their capabilities and enable researchers to solve complex scientific problems more effectively.

## 1.1 Objectives

This project seeks to facilitate the coupling of ScimBa and Feel++. The primary objective is to establish loose coupling between ScimBa and Feel++, enabling efficient utilization of their respective strengths. The general outline as of the start of the project includes the following goals:

1. **Analysis and Planning:** Analyze the functionalities of ScimBa and Feel++ to identify integration points. Develop a plan for implementing loose coupling between the two libraries.
2. **Implementation:** Implement loose coupling between ScimBa and Feel++ according to the planned approach. Develop methods for generating datasets using Feel++ and integrating them into ScimBa.
3. **Testing and Validation:** Test the integrated system to ensure that communication between ScimBa and Feel++ is seamless. Validate the effectiveness of the coupling by solving complex scientific problems.
4. **Documentation and Reporting:** Document the integration process, including interface definitions and communication protocols. Prepare a final report summarizing the project outcomes and lessons learned.

## 1.2 Roadmap



[width=0.7]images/roadmap1.png [width=0.7]images/roadmap2.png

# 2 Getting started

## 2.1 Creating a Docker container

Creating a Docker container and image for the project offers these key advantages:

1. **Portability:** Run the project on any platform supporting Docker.

2. **Isolation:** Avoid conflicts with other software on the host system.
3. **Reproducibility:** Recreate the exact same environment whenever needed.
4. **Dependency Management:** Package all dependencies within the Docker image.

We're using Feel++ as the base for the Docker container adding the requirements and dependencies for Scimba.

This contains the latest version of Feel++ and Scimba and should be able to run these commands without error:

Listing 1: Docker image with Feel++ and Scimba

```

1 python
2 import feelpp
3 import scimba
4
5 FROM ghcr.io/feelpp/feelpp:jammy
6 LABEL maintainer="Helya Amiri<helya.amiri@etu.unistra.fr>,-Rayen-
   Tlili<rayen.tlili@etu.unistra.fr>"
7 LABEL description="Docker image with Feel++ and Scimba."
8 # Install additional dependencies for Scimba
9 RUN apt-get update && apt-get install -y \
10 python3 \
11 python3-pip
12 # Install Scimba directly
13 RUN pip3 install --no-cache-dir scimba
14 # Set the default command to run when the container starts
15 CMD ["python3", "-c", "import feelpp; import scimba"]

```

## 2.2 Exploring Feel++ toolboxes

As of the first meeting with the project supervisors, we've taken a look at the different toolboxes Feel++ has to offer in python:

[width=0.7]images/pyfeelpptoolboxes.png

An interesting toolbox to start with is the **Coefficient Form PDEs**.

[width=0.7]images/cfpde.png

This toolbox allows users to solve PDEs in a generic manner just by inputting the different coefficient values. Implementing an interface which, through Scimba, calls Feel++, solves the PDE and retrieves the solution would be very powerful as it enables us to approach solving for Laplacian and heat transfer problems for example.

### 3 Bibliography

- Feel++ Documentation: <https://docs.feelpp.org/user/latest/index.html>
- ScimBa Documentation: <https://sciml.gitlabpages.inria.fr/scimba/>
- Coupling (Computer Programming): [https://en.wikipedia.org/wiki/Coupling\\_\(computer\\_programming\)](https://en.wikipedia.org/wiki/Coupling_(computer_programming))
- Using feel++: <https://www.cemosis.fr/events/course-solving-pdes-with-feel/>