XIMERA

Interactive Mathematics Education Resources for All

User Manual

Fowler • Obbels • Nowell • Snapp

This document was typeset on 2024/10/06.
GIT COMMIT: 69983ca11a82b4dd01939c742ac25e3efb2be7cf
Source: https://github.com/ximeraProject/ximeraManuals

To lovers of mathematics everywhere.

Contents

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

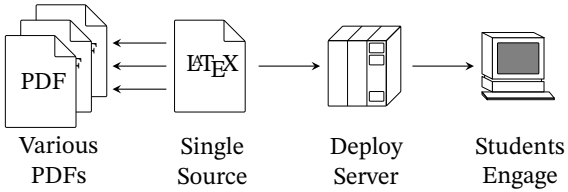
1 Introduction and setup

1.1 About Ximera

What is Ximera? What is it supposed to do? Who is it for?

Ximera, pronounced "chimera," (**X**imera: **I**nteractive, **M**athematics, **E**ducation, **R**esources, for **A**ll) is an open-source platform that provides tools for authoring and publishing (PDF and Online), open-source, interactive educational content, such as textbooks, assessments, and online courses.

Authors write and store their content on their own machines and GitHub repositories. Authors own their content and decide how to license their content. From a single source written in LaTeX, Ximera generates various output: PDF worksheets, PDF textbooks, and PDF solution manuals, and so on. Of most interest, Ximera can also create online interactive activities:



The source code used to produce PDFs can also create interactive online activities when deployed to a Ximera server. Students access this content via a URL or an assignment in their LMS.

Students interact with the *content* produced within Ximera, hence their experience is highly dependent on the *quality* of this content. Research shows that students find Ximera materials to be more readable than traditional course materials and perform equivalently to those using proprietary textbooks and online homework systems. While students typically encounter Ximera through their courses, many discover it via web-search and use the platform as independent learners. In 2023, Ximera has over one million unique visitors. Since Ximera materials are free, they are accessible to anyone, regardless of enrollment in official courses.

Get involved by contributing as an instructor, author, or developer. To get started with Ximera, visit our *First Steps in Ximera* GitHub repository:

https://go.osu.edu/xfs

This document assumes you have completed the instructions there, and have successively deployed Ximera content online.

Funding for the Ximera Project is provided by a U.S. Department of Education Open Textbooks Pilot Program grant in the amount of $2,125,000, from 2024–2026, with no external funding. In the past, the Ximera Project has also received support from NSF Grant DUE-1245433, the Shuttleworth Foundation, the Ohio State University Department of Mathematics, and the Affordable Learning Exchange at Ohio State.

As a token of our appreciation, **consider applying for a Ximera Flash-Grant Stipend:**

https://go.osu.edu/ximera-flash-grant

Thank you for your interest in Ximera. We encourage you to contact the team with any questions you may have.

The authors listed on the cover are the current Ximera lead developers. In reality, this document has many authors as it is part of an evolution of Ximera documentation. Rodney Austin, Oscar Levin, Matt Thomas, and Hans Parshall authored parts of the either the document class or original documentation.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

1.2 Authoring tools

Tools for working on Ximera documents.

Authors own their content in Ximera and can work on their own machine. This has been a fundamental aspect of Ximera since the beginning. To write a Ximera document authors can use their preferred editor, and compile to PDF with a local TeXLive or MikTeX installation that has the Ximera package, as available from CTAN.

However, one needs some extra software to generate and deploy the online version. This is most conveniently provided through Visual Studio Code and Docker. Moreover, the version control system `git` is needed. And, when using Docker a local TeX installation is no longer necessary.

1.2.1 Docker

Docker is a development utility allow you to easily run self-contained applications on your computer. You must install and start Docker before you can deploy. You can just let it run in the background. You can check the status of Docker by running `docker ps` in a terminal session, provided by Visual Studio Code.

1.2.2 Visual Studio Code

Visual Studio Code is a popular text-editor. It is has strong LaTeX, git, and Docker integration. It provides a UNIX-like command line interface on Windows machines via Microsoft's WSL (Windows Subsystem for Linux).

A typical workflow would be to:

(a) Open Docker and minimize the window.

(b) Open Visual Studio Code, do File → Open Folder, and select the folder of your git repository.

(c) To open files, do `Ctrl-p` and start typing file names. Any file committed to your git repository will be found, and files in `.gitignore`, will not be shown.

(d) To run a special command (like search and replace) do `Ctrl-Shift-p`, and search for the command.

(e) To toggle a UNIX-like terminal, use `Ctrl-~`.

Visual Studio Code can be enhanced with a number of extensions. Upon opening the ximeraFirstSteps repo for the first time, you'll be asked to install some extensions (that are needed for the recommended Ximera workflow)

- LaTeX Workshop by James Yu (or LaTeX by Mathematic Inc for a more minimal version)
- Task Buttons by spencerwmiles
- Git

1.2.3 `git`

`git` is program for version control. It allows many authors to work on the same document without fear of overwriting others work. We will discuss `git` in more detail later. It keeps a *complete history* of your work, and can be reverted to a previous state. Generating a PDF from a Ximera document is possible without using git, but in order to generate HTML and to deploy the documents your project has to be stored in a git repository. (Is is not absolutely necessary to publish this repo on GitHub or similar)

Use `git` in Visual Studio Code to add files, commit, push and pull. Alternatively, you can also use the universal and powerful git command line interface from a terminal window, which we'll briefly introduce next.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

If you make a new file, you need to *add* it to the repository. Suppose the name of your new file was MY-NEW-FILE.tex. You should type:

```
git add MY-NEW-FILE.tex
git commit -m "I've add MY-NEW-FILE.tex"
git push
```

The `add` includes the file in the files that `git` tracks. The `commit` says, "hey I made a change that I want to keep track of." The `push` says "I want to share this with others."

If you update a file, type use `git add -u` and it will add all updated files.

```
git add -u
git commit -m "I updated some files"
git push
```

You can combine these commands into one with `&&`:

```
git add -u && git commit -m "this is my change" && git push
```

When you commit you must type a message, describing the commit. You don't have to leave a *good* message, but you must say something. If you just type `git commit` you may end up in `vi`, the terminal will look like:

```
~
~
~
~
~
~
```

In this case, take a deep breath and

press `Esc` then press `:` then press `q` then press `!` then press `Enter`

Then run

```
git config --global core.editor "nano"
```

This will place you in a more user friendly editor in the future.

If you are collaborating, you should always start your work off with

```
git pull
```

typed in terminal. This will retrieve any updates your collaborators have made. Then you do your work, make sure your files compile, then do

```
git add -u && git commit -m "this is my change" && git push
```

Conflicts sometimes occur even when using `git`. When this happens you'll need to *resolve* the conflict. To do this, open the file where the conflict is.

```
git status
```

should tell you this. Once you find the file, open it, and search for "HEAD" and you'll find something like

```
<<<<<<< HEAD
Some stuff that some person wrote.
=======
Some other stuff that someone else wrote
```

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

```
>>>>>>> 3d13c5bba8dc2c12b7783e4f85315b1773165fd6
```

In this case just edit the lines starting with !«««<! and ending with the line >>>>>>>
and make it look how it should. Then do

```
git add -u
git commit -m "I updated some files"
git push
```

and the conflict should be resolved.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

1.3 Set up a GitHub repository

How to set up your Ximera files in GitHub.

All Ximera files must be hosted in a git repository To deploy, you will also need additional files found in `ximeraNewProject`. We suggest you make your own personal copy of `ximeraNewProject`, and build your new Ximera projects directly there. This process is called *forking a GitHub repository*.

1.3.1 Forking `ximeraNewProject`

Forking a repository is well-documented on GitHub[1] . Basically, you login to GitHub, return to this page, and at the top right there will be an option to **fork** this repository. Fork the repo. Accept all defaults, though you want to **change the name of the repository** at this point, we'll use "YOUR-REPO-NAME" for this in the discussion below. When done, it will take you to your copy of this repository on GitHub. It will be located someplace like:

```
https://github.com/YOUR-GIT-USER-NAME/YOUR-REPO-NAME
```

Once the repository is forked, clone the forked repository (the one in your user-space) onto your computer. **If you are using Windows, be sure to clone through WSL.** To do this, open Visual Studio Code, and hit `Ctrl-~` to open a terminal window (select WSL if you are on Windows), and run something like:

```
git clone https://github.com/YOUR-GIT-USER-NAME/YOUR-REPO-NAME
```

or

```
git clone git@github.com:YOUR-GIT-USER-NAME/YOUR-REPO-NAME
```

After the repository is on your computer, you should change the names of a few files. Change the name of `newCourse.tex`, to be a reasonable, web-safe name, say `GOOD-NAME.tex`. To do this run

```
git mv newCourse.tex GOOD-NAME.tex
```

You'll also want to set the license to be something other than the Public Domain. We've included the CC BY-NC-SA 4.0 License as a suggested license; however, you do you. You can either edit this to be another license or move it on to the other license with

```
git mv SUGGESTED-LICENSE.md LICENSE.md
```

Then do

```
git add -u
git commit -m "set course name and license"
git push
```

Now set up (or copy) your `.ximeraserve` file as described in the `ximeraFirstSteps` repository.

To deploy an existing repository, just copy the files you will need to **copy**

- `.gitignore`
- `.vscode`
- `scripts`

[1] See GitHub at https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/working-with-forks/fork-a-repo

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

into your repository. If you already have a `.gitignore` file, we suggest you **replace yours with ours.** Commit your changes to GitHub and view your repository online via a web browser to ensure that the files were added. Finally, after the `.gitignore` is pushed to the repository, set up (or copy) `.ximerserve` to your repository.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

2 Getting work done

2.1 Create an activity

How to set up an activity.

It is easy to write a basic Ximera document with clean LaTeX. The structure is as follows:

```
\documentclass{ximera}
\begin{document}
%%
%% Content goes here
%%
\end{document}
```

Use standard LaTeX environments and commands like: `enumerate`, `itemize`, `description`, `align*`, `$...$`, `\[...\]`, and so on. Macros made of basic LaTeX commands are fine. **No manual formatting.**

You may need to include a `preamble.tex` file. We will discuss that later in this manual.

2.2 Create a xourse

We group Ximera activities into a collection.

Ximera documents can be *glued* together using the `xourse` document class. A `xourse` file is basically a list of other `ximera` files and even other `xourse` files. The `xourse` file for *A First Step in Ximera* can be found at the top level of the GitHub repository `ximeraFirstSteps` here

https://github.com/XimeraProject/ximeraFirstSteps/blob/main/aFirstStepInXimera.tex

The `xourse` documentclass specifies information such as the name of the document, the names of the authors, a description of the content, a license and the names of all Ximera LaTeX files comprising the whole document.

WARNING: All document and folder names used for Ximera must be web-safe! This means all document and folder names:

Must only use alphanumeric English characters Meaning: a,b, …, z, A,B, …, Z, 0,1, …, 9, and hyphen '-' and underscore '_' though the last two are discouraged.

Cannot use use any other characters, including spaces This means all Ximera documents and folders file names must be a single word.

This is not a limitation of Ximera, rather it is a rule that nearly all web-accessible documents must follow.

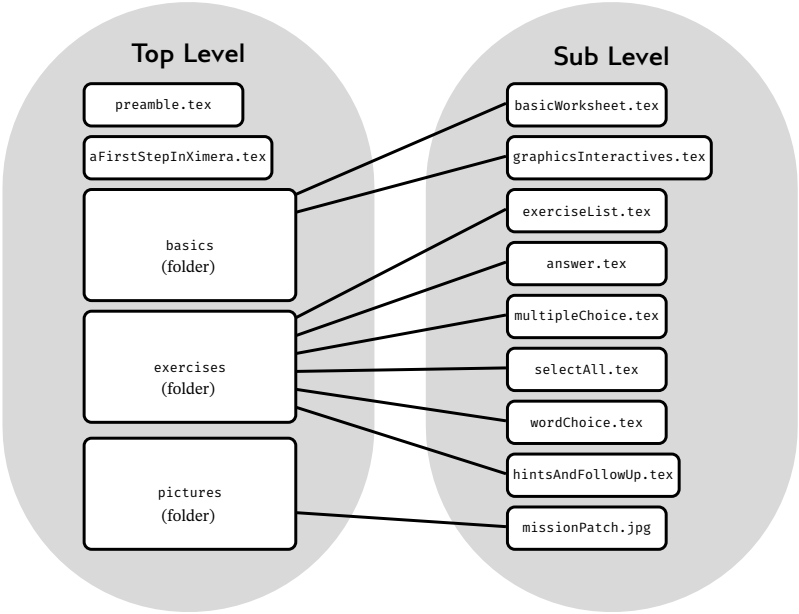
2.2.1 File structure

While one can write a Ximera document without the use of folders, this quickly turns into a mess that is difficult to understand and should only be used for the most basic Ximera content. To help others (including your future-self) work with larger projects, you should have a file structure that helps developers understand where content is stored. We recommend the following:

Group by concept by having all documents that are closely related in idea or scope in the same folder. If someone else wants to use your content, this will be a one-stop destination for them.

Descriptive file names will help you and others understand the structure of your repository. Give you documents descriptive file names like: `completeTheSquare.tex` or `derivativeRules.tex` rather than generic names like: ~~`chapter1.tex`~~, or ~~`math872ch2sec3.tex`~~. Authors find themselves reordering their content and generic names are useless for other users.

For the repository `ximeraFirstSteps` we have the following structure:



We've left some files out of this diagram; regardless, you should be able to goto

https://github.com/ximeraProject/ximeraFirstSteps

and witness this file structure. At the top level of the repository, we have the documents `preamble.tex` and `aFirstStepInXimera.tex` along with the folders `basics` and `exercises`. Inside `basics`, we have two activities and a JPG that is required by one of them. Inside `exercises` we have all the practice exercises.

A consistent and well thought out set up will allow you and others to easily understand and modify your content for years to come. Ideally a document's parent folder would contain everything that document needs to compile, except for perhaps the preamble, and we address this below.

> **WARNING:** Every `*.tex` file in the repository with a `\documentclass` **must compile** for online deployment.

2.2.2 Including with `\activity` and `\practice`

Once you have some files and a basic directory structure, you can add them to the `xourse` document.

> **WARNING:** The `xourse` document will only be easily accessible online if it contains a title, abstract, and `\maketitle` command.
>
> If a `xourse` does not contain a title, abstract, and `\maketitle` it will still be deployed but you will need to know the path to the file. It will be something like:
>
> `https://ximera.osu.edu/YOUR-COURSE-NAME/PATH-TO-YOUR-FILE`
>
> It is important that there is no trailing '/' as
>
> - `https://ximera.osu.edu/moooculus` is correct but
> - ~~`https://ximera.osu.edu/moooculus/`~~ is not.

There are two different commands we use to add `ximera` documents to a `xourse` file:

`\activity` is for including Ximera documents that **include a title, abstract, and `\maketitle`**. They will be represented by their title online. This command is typically used for worksheets and sections of textbooks.

`\practice` is for including Ximera documents that **do not include** a title, abstract, and `\maketitle`. They will be represented by a number based on their order in `xourse` file. This command is used for lists of exercises and problem banks.

In `aFirstStepInXimera.tex` we write

```
\documentclass{xourse}
\input{preamble} %% Loads the graphics path
\author{Wim Obbels \and Bart Snapp}
\title{A First Step in Ximera}
\begin{document}
\begin{abstract}
    A simple collection of Ximera activities,
    to be deployed online.
\end{abstract}
\maketitle
\activity{basics/basicWorksheet}
\activity{basics/graphicsInteractives}
\practice{exercises/answer}
\practice{exercises/multipleChoice}
\practice{exercises/selectAll}
\practice{exercises/wordChoice}
\practice{exercises/hintsAndFollowUp}
\end{document}
```

Note how we give the paths to the exercises, it's the folder name, followed by the document name. The `.tex` can be left on if you like.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

2.3 Preambles, input paths, and graphics paths

How to add extra packages and commands

The Ximera documentclass comes with many packages preloaded: enumitem, titlesec, titletoc, titling, url, xcolor, tikz, pgfplots, fancyvrb, forloop, environ, amssymb, amsmath, amsthm, xifthen, multido, listings, xkeyval, comment, gettitlestring, nameref, epstopdf, hyperref.

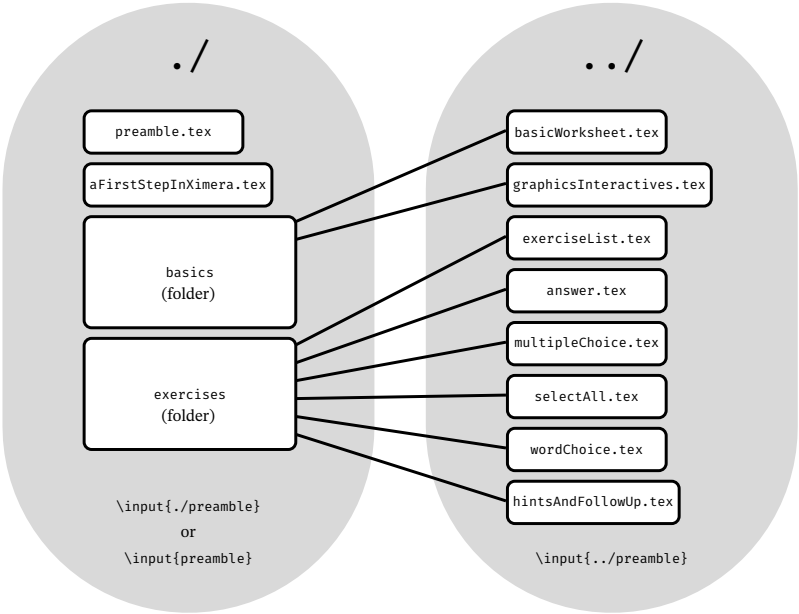
The documentclass also provides support for the following theorem-like environments: algorithm, axiom, claim, conclusion, condition, conjecture, corollary, criterion, definition, example, explanation, exercise, exploration, fact, formula, hypothesis, idea, lemma, model, notation, observation, paradox, proof, problem, procedure, proposition, question, remark, solution, summary, template, theorem, warning.

We can add functionality and generality, via user defined commands, to the document via a preamble, or macros, document. The purpose of a preamble is to ensure **all files compile consistently, it is not for cosmetic changes.** A typically preamble might contain things like:

```
\newcommand{\R}{\mathbb{R}}
\newcommand{\d}{\, d}
```

WARNING: Cosmetic changes to Ximera environments will result in unpredictable behavior online.

Since every *.tex file with a \documentclass must compile, the preamble file must be accessible by all files.



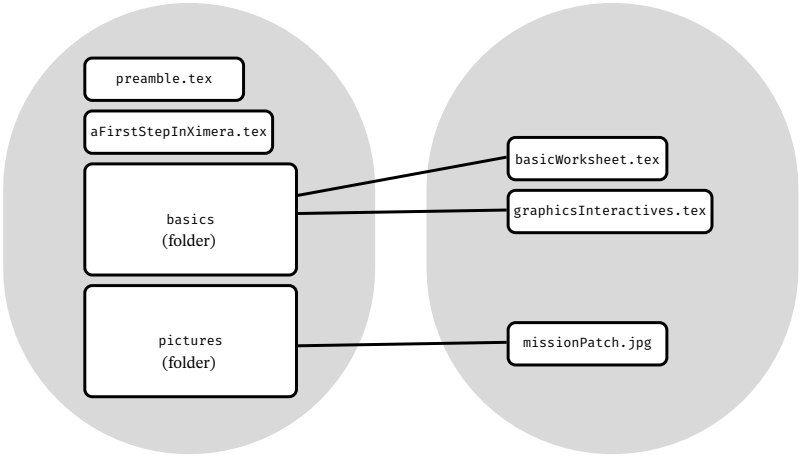
In this diagram, we attempt to show how one should modify the inputs of the preamble based on the directory structure.

Graphics paths help \xourse files find images. We used \includegraphics in basicWorkSheet.tex with code like:

```
\begin{center}
  \includegraphics[width=5cm]{missionPatch.jpg}
\end{center}
```

In this case, basicWorkSheet.tex and missionPatch.jpg are both in the folder basics. However, when this document is compiled via a xourse document, the LaTeX

compiler looks for the image *at the level of the* `aFirstStepInXimera.tex`.



Since there is no file `missionPatch.jpg` (which is on the right) at the level of the `xourse` document (on the left) the compilation will fail. To solve this problem we use the preamble to append the graphics path. If we add (and in reality we did!) this

```
\graphicspath{  %% When looking for images,
{./}            %% look here first,
{./pictures/}   %% then look for a "pictures" folder,
{../pictures/}  %% which may be a directory up.
}
```

to our preamble, then when compiling `aFirstStepInXimera.tex`, the compiler will know where to look.

Input paths allow you to load other documents into your files. This is a less common use case, but we'll mention it here. You can surely avoid the use of such files with an enhanced preamble. However, for this document, we needed to input various TikZ files to help us draw our diagrams. Hence we needed to add the location of these files to our input path. We did this in our preamble the following way:

```
%% Where to look for inputs
\makeatletter     %% make "@" a letter-character
\def\input@path{  %% When looking for files,
{./}              %% look first at your level
{./coverArt/}     %% then in this folder,
{./introduction/} %% then in this folder,
}
\makeatother      %% make "@" an other-character
```

With this said, modifying input paths is more of an advanced feature and one should be careful.

2.4 Graphics

Embed pictures in Ximera activities.

We've seen basic ways to include JPGs, PNGs, and PDFs using `\includegraphics`. However, this is not the preferred way to include graphics. Moreover, there are considerations for positioning of the graphics.

2.4.1 Positioning graphics

In LaTeX it is common to write images in the `figure` environment. We choose not to use this because `figure` 'floats' the images for 'optimal' page layout. We are not concerned with page layout. When working online, the page is essentially infinite in length. Moreover, the *consumers* of the content we create are *students*. Students are also unconcerned with awkward page layout. Students prefer to see the image exactly when it is mentioned. With this said, we suggest wrapping all images in either a `center` environment or an (Ximera-specific) `image` environment. The environment `image` centers and automatically scales the contents. If an author finds themselves printing to various page-sizes, `image` might be preferred. Moreover, `image` can be redefined globally to act identically to `center`, but `center` cannot be redefined.

If you use `center`, you would write something like:

```
\begin{center}
\includegraphics[width=5cm]{missionPatch.jpg}
\end{center}
```

If you use `image`, you would write something like:

```
\begin{image}
\includegraphics[width=5cm]{missionPatch.jpg}
\end{image}
```

The disadvantage of using `\includegraphics` is that you need to handle the paths to the images in some way. In the past we've added the files in the repository to the graphics path. However, we now suggest you use a global graphics path. The repository

https://github.com/XimeraProject/ximeraNewProject

Comes with the file `preamble.tex` with

```
\graphicspath{  %% When looking for images,
{./}            %% look here first,
{./pictures/}   %% then look for a "pictures" folder,
{../pictures/}  %% which may be a directory up.
}
```

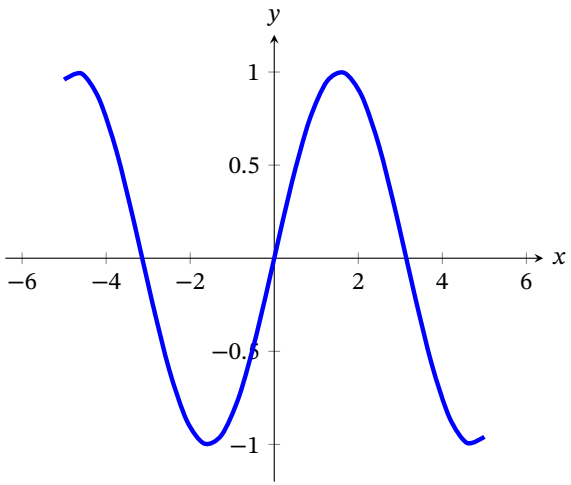
This means that any required `*.jpg`, `*.png`, and `*.pdf` can be found, assuming your file isn't nested too deep. Moreover, we've set up the `.gitignore` to not ignore `*.jpg`, `*.png`, and `*.pdf` when they are placed in the `pictures` directory. Finally, while this is currently the *easiest* method for adding graphics in Ximera, it might be better if all graphics were placed at the level of the source file, and then a soft link (using the terminal) could be made to the `pictures` directory. The advantage here being that if someone wants to borrow your material, everything they need is in the same folder.

With this said, TikZ is the preferred method for graphics because code, found in the Ximera LaTeX file, generates the image. No need to worry about graphics paths.

2.4.2 TikZ is the preferred method for graphics

Most of TikZ is supported, and these are rendered as PNGs online. For example the image below:

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/



was generated via

```
\begin{image}
\begin{tikzpicture}
  \begin{axis}[
  xmin=-6.4,
  xmax=6.4,
  ymin=-1.2,
  ymax=1.2,
  axis lines=center,
  xlabel=$x$,
  ylabel=$y$,
  every axis y label/.style=
    {at=(current axis.above origin),anchor=south},
  every axis x label/.style=
    {at=(current axis.right of origin),anchor=west},
  ]
  \addplot [ultra thick, blue, smooth] {sin(deg(x))};
  \end{axis}
  \end{tikzpicture}
  \end{image}
```

When you use TikZ for graphics, you don't need to worry about `\graphicspath`.

2.5 Desmos and Geogebra

Embed compelling content in Ximera activities.

2.5.1 The graph command

The easiest way to include an interactive graph is to use the `\graph` command. Unfortunately, the `\graph` command doesn't draw a graph in the PDF, rather, it states (in words) that a graph is produced. That is,

```
\[
  \graph{x^2,x^3}
\]
```

produces

Graph of x^2, x^3

There are a number of options concerning the function being graphed:

```
\graph{x^2,x^3}                    %% just x^2 and x^3
                                   %%
\graph{x^2                         %%
\left\{ 1 \leq x \leq 10 \right\}} %% restricted domain
                                   %%
\graph{\sin(x) \left\{x<0\right\}, %%
2x \left\{ x>=0 \right\} }         %% piecewise
                                   %%
\graph{r=\theta}                   %% polar
```

While the code above modifies the function being graphed, there are also several options for the display of the graph.

Optional arguments for `\graph`

`xmin, ymin, xmax, ymax` These set the size of the viewing window with `\graph[xmin=-5,xmax=5,ymin=-5,ymax=5]{y=x^2}`.

`panel` Determines if the panel is shown with `\graph[panel]{y=x^2}`.

`xAxisLabel, yAxisLabel` Gives the axes labels with `\graph[xAxisLabel="time", yAxisLabel="distance"]{y=x^2}`.

`hideXAxis, hideYAxis` Hides the axes with `\graph[hideXAxis=true, hideYAxis=true]{x^2}`.

`hideXAxisNumbers=true, hideYAxisNumbers=true` Hides the tick marks on the axes with `\graph[hideXAxisNumbers=true, hideYAxisNumbers=true]{y=x^2}`.

`polar` Shows polar grid lines with `\graph[polar]{y=x^2}`.

Graph of $x^2\{1 \leq x \leq 10\}$Graph of $\sin(x)\{x < 0\}, 2x\{x >= 0\}$

2.5.2 Desmos, Desmos 3D, and GeoGebra

If you require further features from Desmos[2] , you can sign up for an account and include your worksheets using the syntax `\desmos{ID}{width}{height}`, where ID is the widget ID and width and height are the dimensions (in pixels) you want the embedded widget to have.

```
\begin{center}
\desmos{zwywds7med}{800}{600}
\end{center}
```

which renders as:

Desmos link: https://www.desmos.com/calculator/zwywds7med

[2] See Desmos at https://www.desmos.com/

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

The syntax for Desmos 3D is similar. Use `\desmosThreeD{ID}{width}{height}`, where ID is the widget ID and width and height are the dimensions (in pixels) you want the embedded widget to have.

```
\begin{center}
\desmosThreeD{bb4exrhrl3}{800}{600}
\end{center}
```

Seen here:

Desmos3D link: https://www.desmos.com/3d/bb4exrhrl3

You can also use GeoGebra[3] . Embed the widget using the syntax `\geogebra{ID}{width}{height}`, where ID is the widget ID and width and height are the dimensions (in pixels) you want the embedded widget to have.

```
\begin{center}
\geogebra{XC3FXUdJ}{800}{600}
\end{center}
```

Geogebra link: https://www.geogebra.org/m/XC3FXUdJ

While we cannot get data from these sorts of interactives directly, the clever author can ask questions that **use** the interactive to find a solution.

[3] See GeoGebra at https://www.geogebra.org/

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

2.6 Videos and other interactives

Embed compelling content in Ximera activities.

2.6.1 Videos

We can embed YouTube Videos using the syntax `\youtube{ID}`, where ID is the video ID found at the end of the YouTube link after "watch?v=".

```
\begin{center}
\youtube{FvgF95i0_lw}
\end{center}
```

which would embed the video into the page, like this:

YouTube link: https://www.youtube.com/watch?v=FvgF95i0_lw

WARNING: YouTube videos count toward the completion of the activity. As such, currently, students must watch the **entire video to earn full credit**.

XIMERA FORM 8654 GIT COMMIT 69983ca11a82b4dd01939c742ac25e3efb2be7cf PREVIOUS EDITION IS OBSOLETE.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

2.7 Deploying Ximera documents

Deploying Ximera documents.

Finally to deploy,

(a) Make sure all source files are committed and pushed to the repository. A quick

```
git add -u && git commit -m "this is my change" && git push
```

may help. Also, you can check your personal GitHub page to ensure files are in the repository.

(b) Ensure Docker is running.

(c) Press *Bake* (typically on the bottom taskbar in VS Code).

(d) Press *Serve* (typically on the bottom taskbar in VS Code).

We should discuss this more once the discussion in the repository `ximeraFirstSteps` is steady.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

2.8 Common issues

Suggestions for dealing with common issues.

If the Bake fails ensure all files are committed to the repository (verify online when in doubt), ensure all files compile. In particular, ensure filenames are web-safe and all necessary paths are correct. You can check against TeXLive 2019 by running the following:

```
./scripts/xmlatex -i bash
```

in a terminal window, note **you must be at the top level of your repository**. Then you can use `pdflatex` to compile and use

```
pdflatex FILE-NAME.tex
xake -v compile FILE-NAME.tex
more FILE-NAME.tex.log
```

If you find a file that does not compile, you can try to compile the file directly while slowly moving `\end{document}` down through the document. By starting at the top, the file should compile, and then you should be able to locate the exact location of the bug.

An svg-viewer missing error message means that there was a compilation error that confused Xake. The offending file needs to be recompiled. Make a trivial change in the file, and delete all SVG files in the directory.

Sometimes Ximera materials look good to instructors and others, but not to some individual students.

Ximera refreshes the page when submitting can be caused by several issues, each with a different solution:

Problem: Poor internet connection
Solution: Refresh the page immediately before entering each answer.

Problem: Browser settings
Solution: Try using a different browser. Different students have reported luck with different browsers, including Edge, Firefox, and Chrome.

Problem: Antivirus Software
Solution: Some antivirus software (such as McAfee) are now blocking websockets, a technology Ximera relies on. Try turning off the websockets setting on the antivirus software.

Problem: VPN
Solution: If you are using a VPN, try turning it off before using Ximera.

unknown node type: parser error message. This appears sometimes rather than the actual rendered mathematics. This is due to a corruption in the cookies/cache for the browser, and the person seeing this error need to clear their cache and then reload the page to resolve the problem.

Difficulty accessing Ximera is often caused by browser settings. The easiest solution is usually to try using a different browser. Students seem to have the best luck with Chrome, Firefox, or Safari.

Math processing error This is caused by an error in the browser setting. Two things to try:

First possible solution: Clear browser cookies. Close the browser. Open the browser and log in to the LMS (if applicable). Go to the assignment.

Second possible solution: Find one of the pages this is happening, Put your mouse over the ‘math processing error’ Now, ‘right-click’ and select ‘accessibility’ If collapsible math is checked, uncheck it. Reload the page.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

3 Answerables and progress

3.1 Enter math answers with \answer

Let students enter math expressions.

There are many ways our friends use Ximera. Some make worksheets, some make exercise banks, and some make full textbooks. Common to all of these are the use of interactive elements that students submit answers to. We'll call anything a student can *answer* an **answerable**. With that said, we'll discuss the basic answerables below, and then describe how they might be used in use cases of: worksheets, exercise banks, and textbooks.

The basic way of including a answerable item in Ximera is to use the `\answer` command.

WARNING: The `\answer` command must be in **math-mode** (display or inline) and also be **inside of a theorem-like environment**.

Here are some examples of how to use `\answer`. We can validate a single number:

```
\begin{problem}
State the answer to life, the universe, and everything.
\[
\answer{42}
\]
\end{problem}
```

We can validate an expression of variables:

```
\begin{exercise}
Compute:
\[
\frac{\partial}{\partial x} \sin(3xyx)
= \answer{3yz\cos(3xyz)}
\]
\end{exercise}
```

We can validate with several answerables in an environment:

```
\begin{question}
For each of the following functions, find $x$ such that it is a
critical point, or write ``NA'' if none exists.
\begin{enumerate}
  \item For $f(x) = |x-4|$, $x$ is a critical point when
  $x=\answer{4}$.
  \item For $g(x) = x^2-4x+1$, $x$ is a critical point when
  $x=\answer{2}$.
  \item For $h(x) = 3x-2$, $x$ is a critical point when
  $x=\answer[format=string]{NA}$.
\end{enumerate}
\end{question}
```

There are also a number of optional arguments that can be passed to the `\answer` command.

`tolerance` sets a $\pm$ value on the author's (numeric) answer that will be accepted from a student. For example, the answer box `\answer[tolerance=5]{100}` will accept anything in the range of $[95, 105]$ as correct.

`format=string` validates *words* typed into an answer box. Here `$\answer[format=string]{Cat}$` will validate `Cat`, `cAt`,`caT`, `cat` all as correct. However, if spaces are inserted before, between, or after, it will be marked as incorrect.

given will show an answer in the PDF. Sometimes it is necessary to show such content to ensure understandable in a PDF setting. Here *given* means that the answer is *given* to the student in the PDF.

There are two other optional arguments `validator` and `id`, but they are for building custom validators and are beyond the scope of this document.

WARNING: You separate options in `\answer` using commas and there can be no spaces, like this:

```
\[
\answer[given,tolerance=5]{100}
\]
```

Validating an answer is achieved by client-side JavaScript. Around one dozen algorithms check if the student's provided answer is mathematically equivalent to the content author's provided answer. If any report the answer as being correct, then it is marked as correct.

WARNING: There are confounds for validating answers online:

Student input answers a nontrivial task. Students need to be able to easily type in their answers.

Checking for equality will not reveal the form of the answer. This means that sometimes the *answer* to the question can be the question itself. For example `1+2 = \answer{3}` can be answered with '$1+2$' because that *equals* 3.

Right-click reveals the answer to the students. Moreover, the source is open to the students. Ximera is not designed for high-stakes assessment.

Richardson's theorem states it is not possible[a] to decide equality between the expressions of involving integers, π, logarithms and exponential and sine functions.

Floating-point numbers can lead to erroneous computations, leading to identifying nonequal expressions as equal.

[a] See states it is not possible at https://en.wikipedia.org/wiki/Richardson%27s_theorem

In abstract, each of these confounds are quite daunting. However, in practice, they can be mitigated via Ximera's design along with thought on the author's side when constructing problems.

Student input is reflected back to the student, in the sense that Ximera tells the users what it thinks they are typing. In addition, we provide a math palette. Authors can assist students with input by ensuring that the required answer is not too difficult to type in.

Ximera checks for equality and is comprehensive in practice, but that can be problematic when you want the answer in a specific *form*. For example if you want students to sum numbers, you will have to use

```
\[
1+2 = \answer[format=string]{3}
\]
```

For other specific forms, like factoring polynomials, you will need to use a custom validator.

Ximera is open-source and not designed for high-stakes assessment. All content is open and students can find the source if they look hard enough. Moreover, answers can be exposed via right-click. This has the even more unfortunate side-effect that a student can mess up their browser settings to the point that Ximera content will not render.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

For accessibility reasons (like screen readers and the like) the library that powers the answer box has a bunch of features bundled in, one of which is the ability to **right-click and get different formatting** of its contents. Unfortunately, if you right-click and go to "Show Math As" and then "TeX Commands" you can **very easily see the contents of the answer command**.

This information isn't necessary for accessibility (after all, the answer box is supposed to contain the actual content supplied by the student, and doesn't have any content the student needs to know about) but it's tied to the behavior of the underlying library that supports *all* the rendered math on the page, so we can't remove it without killing *all* accessibility…yet.

However, it is easily countered in *specific cases* of the answer box. To do this, in your preamble, add something like:

```
\newcommand{\myHiddenAns}{6}
```

then in your document, you can write

```
\begin{exercise}
  Computer $3!$
  \[
  3! = \answer{\myHiddenAns}
  \]
\end{exercise}
```

Ricardson's theorem forces the developers of Ximera to make a decision:

- To sometimes count correct submissions as incorrect.
- To sometimes count incorrect submissions as correct.

For the Ximera platform, we chose to **sometimes count incorrect submissions as correct**. This means that at least one of the validators believed that the student typed in the correct solution. The validator can surely be optimized further, but that will be for future research and development.

Floating-point computations are a fact-of-life when working with computers.

```
\begin{example}
Confirm:
\[
10 \ne \answer{10^{-11}+10}
\]
\end{example}
```

Ximera accepts 10 for the answer that should be $10^{-11} + 10$ as correct, even though we obviously know that's not true. This because 10^{-11} is a number that is too small for machine precision. For a more in-depth review of the perils of floating-point arithmetic, see *What Every Computer Scientist Should Know About Floating-Point Arithmetic*[4] .

Parenthesis around answer boxes are good to use, especially if your answer is negative and part of an expression.

Since the answer box is a different size than the answer you should use `\left` and `\right` to do it like this:

```
\begin{exercise}
  3 + \left(\answer{-2}\right) = 1
\end{exercise}
```

and here we see this in a more sophisticated context:

[4] See *What Every Computer Scientist Should Know About Floating-Point Arithmetic* at https://docs.oracle.com/cd/E19957-01/806-3568/ncg_goldberg.html

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

```
\begin{exercise}
Given a normal vector $\vec{n} = (a,b,c)$ and a point
$(x_0,y_0,z_0)$, the equation of the plane with normal
vector $\vec{n}$ that passes through $(x_0,y_0,z_0)$ is
given by
\[
  a(x-x_0)+b(y-y_0)+c(z-z_0) = 0.
\]
Find the equation of a plane with normal vector $(2,-1,9)$
that passes through $(2,-3,4)$. Express your final answer
in the form $ax+by+cz=d$.
\[
  2x+\left(\answer{-1}\right)y+\left(\answer{9}\right)z
  = \answer{43}.
\]
\end{exercise}
```

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

3.2 Select correct answers with \choice

Let students select correct answers.

There are three similar answerables that all use the command `\choice`.

WARNING: All answerables must be **inside of a theorem-like environment**.

In each case that uses `\choice`, the order of the choices presented to the student is the order the author types in the code. The author marks correct answers with the option `correct` and leaves incorrect answers without options.

```
\choice[correct]{SOME-CORRECT-ANSWER}
\choice{SOME-INCORRECT-ANSWER}
```

Now we will discuss each specific environment that uses `\choice`.

Multiple choice questions, including True/False, are intended for students to select one correct answer.

```
\begin{question}
Which of the following functions has a graph which is a parabola?
\begin{multipleChoice}
  \choice[correct]{$y=x^2+3x-3$}
  \choice{$y = \frac{1}{x+2}$}
  \choice{$y=3x+1$}
\end{multipleChoice}
\end{question}
```

If more than one choice is labeled correct with `multipleChoice`, any correct answer will result in completion of this answerable.

```
\begin{problem}
Select a prime number:
\begin{multipleChoice}
  \choice{1}
  \choice[correct]{2}
  \choice[correct]{3}
  \choice{4}
  \choice[correct]{5}
\end{multipleChoice}
\end{problem}
```

With `multipleChoice` the student is only able to select one answer before submitting. This could be useful for student surveys where every choice is marked as correct.

Select all problems allow the student to select any and all answers before submitting.

```
\begin{problem}
Select all prime numbers:
\begin{selectAll}
  \choice{1}
  \choice[correct]{2}
  \choice[correct]{3}
  \choice{4}
  \choice[correct]{5}
\end{selectAll}
\end{problem}
```

Select all problems can be very challenging for students. Authors can quickly make questions that are quite difficult without realizing it.

Word choice problems were designed for inline *words*; however, at this point we do support math in the choices. We give an example of `\wordChoice` in action below:

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

```
\begin{exercise}
 Consider the planes defined by the equations below.
\begin{align*}
  P_1:  \quad 4 &= 2x-y+3z  \\
  P_2:  \quad 5 &= 4x-2y+6z \\
  P_3:  \quad 7 &= 5x+2y+z
\end{align*}
Describe the relationships between the planes
$P_1$, $P_2$, and $P_3$ in terms of ``parallel,''
``orthogonal,'' or ``neither.''
\begin{enumerate}
  \item The planes $P_1$ and $P_2$ are \wordChoice{
    \choice[correct]{parallel}
    \choice{orthogonal}
    \choice{neither parallel nor orthogonal}
    }
  \item The planes $P_1$ and $P_3$ are \wordChoice{
    \choice{parallel}
    \choice{orthogonal}
    \choice[correct]
    {neither parallel nor orthogonal}}.
  \item The planes $P_2$ and $P_3$ are \wordChoice{
    \choice{parallel}
    \choice{orthogonal}
    \choice[correct]{neither parallel nor orthogonal}
    }
\end{enumerate}
\end{exercise}
```

It is difficult to have a PDF version of `\wordChoice` (unless you use the documentclass option `wordchoicegiven` so authors should take this into consideration.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

3.3 Problem environments and nesting

One thing you might want to have are problems that unfold as students work. We have special environments for this.

While **any** environment can contain the command `\answer`, there are four special environments: `question`, `exercise`, `problem`, `exploration`. If these environments are nested within each other, They *hide* the inside environment. For example

Problem 1 *Start with* $F_0 = 1$ *and* $F_1 = 1$. *Define*

$$F_{n+1} = F_n + F_{n-1} \qquad \textit{for } n \geq 1$$

Find F_2.

$$F_2 = \ldots\ldots$$

Problem 1.1 *Find* F_3

$$F_3 = \ldots\ldots$$

With this said, this technique should be used with care and perhaps even sparingly. When problems unfold, students don't have any idea when the *pain* will end. If we consider the unfolding problems above, but simply listed in order:

```
\begin{problem}
Start with $F_0 = 1$ and $F_1=1$. Define
\[
  F_{n+1} = F_n + F_{n-1} \qquad\text{for $n\ge$ 1}
\]
Find $F_2$.
\[
  F_2 = \answer{2}
\]
\end{problem}
\begin{problem}
Find $F_3$
\[
  F_3 = \answer{5}
\]
\end{problem}
```

little is lost pedagogically and a student has a clear end in sight.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

3.4 Progress and credit

How Ximera assigns progress to students as they complete an assignment.

For each distinct URL assigned, including an entire `xourse` file, Ximera reports a number between 0 and 1, with 1 representing "complete."

3.4.1 Progress within a Ximera file

The total progress of a given Ximera file is broken up evenly with top level environment and any YouTube videos, which *could* be in environments. For example, let's say you have the following Ximera document:

```
\documentclass{ximera}
\title{An Example Document}
\author{Jane Doe}
\begin{document}
\begin{abstract}
An example to help understand progress.
\end{abstract}
\maketitle

\begin{theorem}
If $x$ is a real number with $x=1$, then $x+x = 2$.
\end{theorem}

\begin{question}
In order to apply the theorem, $x$ must be
(select all that apply):
\begin{selectAll}
    \choice{A variable.}
    \choice[correct]{A real number.}
    \choice{An arbitrary constant.}
    \choice[correct]{Equal to $1$.}
    \choice{Trick question, $x$ is a letter.}
    \end{selectAll}
    \begin{problem}
    What does the theorem conclude that $x+x$ equals?
    \[
    x + x = \answer{2}
    \]
    \end{problem}
\end{question}

\begin{remark}
We have the theorem given to us, and it has two parts,
the ``if'' statement that sets the necessary hypotheses
to apply the theorem, and the ``then'' statement which
tells us the result.
\end{remark}

\begin{problem}
Compute $1+2$.
\[
1+2=\answer{3}
\]
\end{problem}

\begin{center}
    \youtube{FvgF95i0_lw}
\end{center}
\end{document}
```

Above there are four top level theorem environments above: A `theorem`, `question`, `remark`, and `problem`; and a YouTube video. Each of these is worth 20% of the total credit. The breakdown of points is described below:

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

Theorem environment Worth 20% contingent on completion.

No answerables Automatically flagged as complete.

Question environment Worth 20% contingent on completion.

selectAll Sub-environment worth 50% of the enclosing environment contingent on completion.

Problem environment Sub-environment Worth 50% of the enclosing environment contingent on completion.

Remark environment Worth 20% contingent on completion.

No answerables Automatically flagged as complete.

Problem environment Worth 20% contingent on completion.

YouTube video Worth 20% contingent on *complete* viewing.

In table form:

Environment	Problem Level Credit	Total Credit
Theorem		20%
Question		
• selectAll	50%	10%
• Sub-Problem		
• • Answer box	50%	10%
Remark		20%
Problem		
• Answer box	100%	20%
YouTube		20%

There are things to notice here.

(a) Environments that don't have answerables automatically give credit as soon as the page is loaded. This also means that a large portion of the tile's credit could (theoretically) be given just for loading the page, without doing anything. Getting some credit for loading the page doesn't seem to result in less effort spent on the page.

(b) Only environments and YouTube give credit. All answerables **must** be in an environment.

(c) Since credit is split evenly **at the top-level environment count only**, problems with lots of parts, nested or not, can end up with nested problems that are worth significantly less compared to a problem that has no nested problems.

(d) YouTube videos do not need to be in theorem-like environments and one must watch them in entirety to receive full credit.

3.4.2 Progress within a xourse

Credit for a xourse is split evenly between all Ximera activities listed in the xourse.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

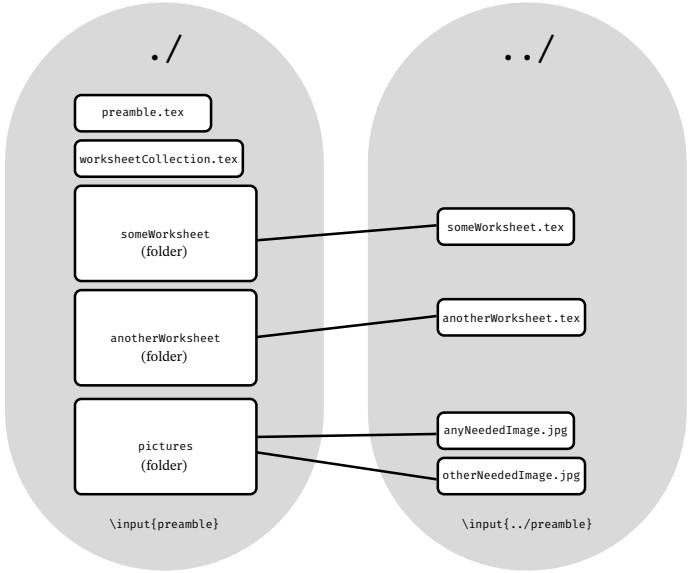
4 Common use cases

4.1 Worksheets

We discuss the set up for a collection of worksheets.

A worksheet is a piece of paper with questions on it. A Ximera worksheet is no different.

File structure



In this setting, the file `worksheetCollection.tex` might look something like

```
\document{xourse}
\input{preamble}
\title{My Groovy Worksheets}
\begin{abstract}
  Some of my worksheets.
\end{abstract}
\maketitle
\begin{document}
\activity{someWorksheet/someWorksheet}
\activity{anotherWorksheet/anotherWorkSheet}
\end{document}
```

The preamble would include:

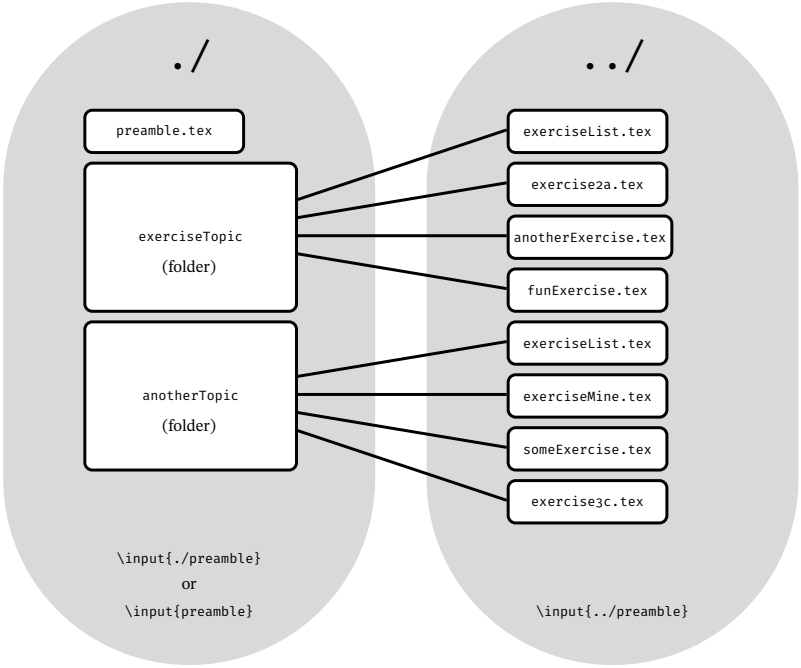
```
\graphicspath{  %% When looking for images,
{./}            %% look here first,
{./pictures/}   %% then look for a "pictures" folder,
{../pictures/}  %% which may be a directory up.
}
```

4.2 Exercise banks

Ximera can present exercise banks.

Exercise banks can be made with `xourse` files. If you are making an exercise bank, you might want to have essentially one exercise per file.

File structure



The individual exercises that live in folders might look something like:

```
\documentclass{ximera}
\begin{document}
\begin{exercise}
Compute:
\[
  \frac{\partial}{\partial x} \sin(3xyx)
  = \answer{3yz\cos(3xyz)}
\]
\end{exercise}
\end{document}
```

In this case, the `exerciseList` within `exerciseTopic` might look like

```
\documentclass{xourse}
\input{../preamble}
\title{My exercise banks}
\begin{document}
\begin{abstract}
Here is an exercise bank
\end{abstract}
\maketitle

\practice{exercise2a}
\practice{anotherExercise}
\practice{funExercise}

\end{document}
```

If you don't give your `xourse` file a title, it will not show up on the front page. Instead you can link directly to the file by going to something like:

`https://some.ximera.server/DEPOLOY-NAME/exerciseTopic/exerciseList`

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

Exercises as part of a course can be included as a folder within the section that the exercises address. The next section will address this in detail.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

4.3 Textbooks

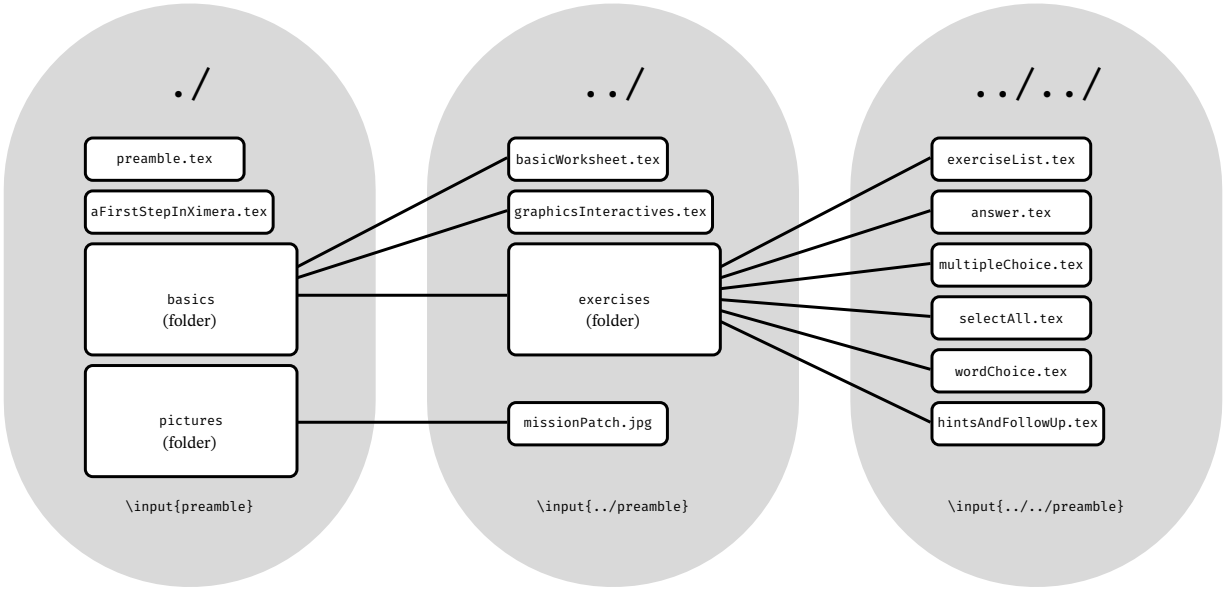
You can write textbooks in Ximera.

This section is about *best practices* when writing an entire textbook.

Keep related content in the same folder to help yourself and others find content. For example, it is helpful to keep exercises that complement a section of a textbook in the same folder as that section.

Give descriptive names to help others and your future self find things. Abstract names like `chapter1` and `course` are too generic, and are at best unhelpful and at worst confusing when things are rearranged.

Pay attention to file paths as every file with a documentclass must compile, which could mean that files need to find the `preamble.tex` file.



Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

To help organize your book, you should use the documentclass option `numbers` along with `\chapterstyle` and `\sectionstyle`. Keep your sections short and sweet. Two things that authors commonly do are:

- When you give a definition, ask a question after to check comprehension.
- When you give an example, give an explanation with `\answer` boxes to fill in.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

5 Further information

5.1 Printing with style

We describe methods of styling the print versions of Ximera materials.

5.1.1 Documentclass options

There are a number of options for the document class, though their effects are only seen in the PDF:

handout The default behavior of the class is to display **all** content. This means that if any questions are asked, all answers are shown. Moreover, some content will only have a meaningful presentation when displayed online. When compiled without any options, this content will be shown too. This option will supress such content and generate a reasonable printiable "handout."

noauthor By default, authors are listed at the bottom of the first page of a document. This option will supress the listing of the authors.

nooutcomes By default, learning outcomes are listed at the bottom of the first page of a document. This option will supress the listing of the learning outcomes.

instructornotes This option will turn on (and off) notes written for the instructor.

noinstructornotes This option will turn off (and on) notes written for the instructor.

hints When the `handout` option is used, hints are not shown. This option will make hints visible in handout mode.

newpage This option will start each problem-like environment (`exercise`, `question`, `problem`, and `exploration`) on a new page.

numbers This option will number the titles of the activity. By default the activities are unnumbered.

wordchoicegiven This option will replace the choices shown by `wordChoice` with the correct choice. No indication of the `wordChoice` environment will be shown.

5.1.2 The preamble versus printing styles

The preamble is used to make the Ximera document compile. In fact, it may be the case that some files don't even need the preamble. When this is the case, you don't need to worry about it.

The printing style is used to make cosmetic changes to the PDF. They are only included in the `xourse` files and are enclosed in `\pdfOnly`. For example the printstyles for this document are loaded with

```
\pdfOnly{\usepackage{manual}}
```

5.1.3 When and how to use `prompt`

Prompt allows you to make your exercises look beautiful in the PDF. As an example consider this exercise:

```
\begin{exercise}
Compute $\frac{d}{dx} x^2$
\begin{prompt}
\[
  \frac{d}{dx} x^2 = \answer{2x}
\]
\end{prompt}
\end{exercise}
```

It renders as

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

Exercise 2 *Compute* $\frac{d}{dx}x^2$

The environment `prompt` hides the display math in the PDF. In a PDF, you don't need answer boxes! As another example, consider:

```
\begin{exercise}
 Consider the planes defined by the equations below.
\begin{align*}
  P_1:  \quad 4 &= 2x-y+3z  \\
  P_2:  \quad 5 &= 4x-2y+6z \\
  P_3:  \quad 7 &= 5x+2y+z
\end{align*}
Describe the relationships between the planes
$P_1$, $P_2$, and $P_3$ in terms of ``parallel,''
``orthogonal,'' or ``neither.''
\begin{prompt}
\begin{enumerate}
  \item The planes $P_1$ and $P_2$ are \wordChoice{
    \choice[correct]{parallel}
    \choice{orthogonal}
    \choice{neither parallel nor orthogonal}
    }
  \item The planes $P_1$ and $P_3$ are \wordChoice{
    \choice{parallel}
    \choice{orthogonal}
    \choice[correct]
    {neither parallel nor orthogonal}}.
  \item The planes $P_2$ and $P_3$ are \wordChoice{
    \choice{parallel}
    \choice{orthogonal}
    \choice[correct]{neither parallel nor orthogonal}
    }
\end{enumerate}
\end{prompt}
\end{exercise}
```

The environment `prompt` says that the stuff inside the environment is used as a *prompt* for the student. In the PDF, everything within the prompt is hidden, as a student doesn't need that content to answer the question using pencil and paper.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

5.2 Rendering content

A description of how text, math, images, and interactive content are rendered.

Rendering in Ximera can be broken down into three parts:

Rendering text is handled using the LaTeX package TeX4ht.

Rendering math is handled using MathJax via TeX4ht.

Rendering images is done by directly showing the image (in the case of a PNG or JPG) or converting to SVG and displaying the SVG.

5.2.1 Basics of rendering

There are issues with variables like “textwidth” versus “pagewidth” used in LaTeX. For online rendering, these variables are all basically made to be the width of the browser window (more or less—there are some very technical details here). This can make it difficult to horizontally align things with any subtlety.

5.2.2 Accessibility

MathJax has extensive accessibility features built in, which means Ximera benefits from the developers keeping this up-to-date and conforming to industry standard. In essence, you don’t need to worry about accessibility features for rendered math content - with the exception of graphs.

Due to how graphs are rendered, they currently don’t have any accessibility features (e.g. alt-text) if you provide them via TikZ or other LaTeX means. You can input them as image files instead, however those also lack any accessibility support. This is something to keep in mind, as you may need to provide textual description explicitly for things like screen readers to provide more accessibility for graphs and/or images.

Email: ximera@math.osu.edu Website: https://github.com/ximeraProject/

5.3 See Ximera in action

Examples of documents in Ximera.

Precalculus Algebra	(https://go.osu.edu/prca)
ComprehensiveFactoring Practice Quiz	(https://go.osu.edu/cfa)
Precalculus with Review: Part I	(https://go.osu.edu/pcr)
Precalculus with Review: Part II	(https://go.osu.edu/pcr2)
Business Calculus	(https://go.osu.edu/bca)
Calculus 1	(https://go.osu.edu/mc1)
Calculus 2	(https://go.osu.edu/mc2)
Calculus 3	(https://go.osu.edu/mc3)
Linear Algebra: An Interactive Introduction	(https://go.osu.edu/ila)
Systems: Number of Solutions (with Parameter)	(https://go.osu.edu/mepr)
Math for Elementary Teachers: Part I	(https://go.osu.edu/met1)
Math for Elementary Teachers: Part II	(https://go.osu.edu/met2)
Quality Control for Introductory Statistics and Manufacturing Courses	(https://go.osu.edu/qcst)

To see the code on any Ximera webpage, simply append .tex to the URL after loading the page. Enjoy!

XIMERA FORM 8654 GIT COMMIT 69983ca11a82b4dd01939c742ac25e3efb2be7cf PREVIOUS EDITION IS OBSOLETE.

Ximera, pronounced “chimera,” (Ximera: Interactive, Mathematics, Education, Resources, for All) is an open-source platform that provides tools for authoring and publishing (PDF and Online), open-source, interactive educational content, such as textbooks, assessments, and online courses.
Authors write and store their content on their own machines and GitHub repositories. Authors own their content and decide how to license their content. From a single source written in LaTeX, Ximera generates various output: PDF worksheets, PDF textbooks, and PDF solution manuals, and so on. Of most interest, Ximera can also create online interactive activities:

The source code used to produce PDFs can also create interactive online activities when deployed to a Ximera server. Students access this content via a URL or an assignment in their LMS.

This document was typeset on 2024/10/06 10:07:18.
GIT COMMIT: 69983ca11a82b4dd01939c742ac25e3efb2be7cf
Source: https://github.com/ximeraProject/ximeraManuals