

# emo•ji for all (LaTeX engines)

Robert Grimm

Version v1.0 (2023/99/99)

## Abstract

Emo implements the `\emo{<emoji-name>}` command for including color emoji such as `\emo{desert-island}` for  or `\emo{parrot}` for  in your documents independent of LaTeX engine. The implementation uses the Noto color emoji font if the engine supports it and includes PDF graphics otherwise. It also supports conversion to HTML with either LaTeXML or TeX4ht. Next, PDF graphics are automatically derived from Noto's SVG sources, so the visual appearance is very similar. The source repository is at <https://github.com/apparebit/emo>. Emo may come in particularly handy when dealing with academic publishers that provide only minimal support for non-Latin scripts (cough, ACM, cough).

## Contents

<b>1</b>	<b>Installation</b>	<b>2</b>
<b>2</b>	<b>Usage</b>	<b>3</b>
2.1	One Main Macro . . . . .	3
2.1.1	Naming Scheme . . . . .	3
2.1.2	Inventory of Default Configuration . . . . .	5
2.2	Two Optional Macros . . . . .	20
2.3	Conversion to HTML . . . . .	20
<b>3</b>	<b>Configuration</b>	<b>20</b>
3.1	Running the Configuration Script . . . . .	20
<b>4</b>	<b>Copyright and Licensing</b>	<b>21</b>
<b>5</b>	<b>Implementation</b>	<b>21</b>
5.1	Package Options . . . . .	22
5.2	Package Dependencies and Backend Selection . . . . .	22
5.3	The Emoji Table . . . . .	23
5.4	Internal Macros . . . . .	24
5.5	User Macros and Hooks . . . . .	25
5.5.1	Background: Variability vs Code Duplication . . . . .	25
5.5.2	A Skeleton of Render Hooks . . . . .	26
5.5.3	The User Macros . . . . .	27
5.5.4	Activating the Hooks . . . . .	27

<b>6</b>	<b>LaTeXML Binding</b>	<b>28</b>
<b>7</b>	<b>Testing and Documenting Emo</b>	<b>30</b>
7.1	Emo's Support Package . . . . .	30
7.1.1	Report Generation . . . . .	31
7.1.2	Emoji Inventory . . . . .	32
7.1.3	Testing . . . . .	34
7.2	Engine Test and Report . . . . .	35
7.3	Simple Test Document . . . . .	36

## 1 Installation

The `emo` package is available through its [source repository](#) or through [CTAN](#). Installation is fairly straightforward, though it does involve a lot more files than usual.

1. Start by extracting this package's files from `emo.dtx` by running:

```
$ pdflatex emo.dtx
```

Do *not* use `tex`; it mangles the embedded `README.md`. `pdflatex` also extracts the files and then builds the documentation. Embedded files are `build.sh`, `emo.ins`, `emo.sty`, `emo.sty.ltxml`, `emo-support.sty`, `canary.tex`, `demo.tex`, and `README.md`. Extraction will overwrite existing files with the same name without asking.

2. Test the package and build its documentation by making `build.sh` executable and then running it:

```
$ chmod +x build.sh; ./build.sh
```

The shell script does three things: First, it tests `emo` by compiling `canary.tex` with pdfTeX, XeTeX, as well as LuaTeX and generating `canary.pdf` with the results from the three tests. Second, it tests `emo` by compiling `demo.tex` with LaTeXML as well as TeX4ht and generating `demo-latexml.html` as well as `demo-tex4ht.html`, which are framed together by `demo.html`. Third, it builds the package documentation in `emo.pdf` with all bells and indices.

3. Get started reconfiguring supported emoji by running:

```
$ python config/emo.py -h
```

For more detailed instructions, see §3 below.

4. Put the following files somewhere LaTeX can find them. In a pinch, your current project's directory will do. However, `emo`'s installation potentially comprises thousands of files. So, you probably want to use a dedicated directory and add that to the search path for LaTeX, e.g., by setting the `TEXINPUTS` environment variable.

- (a) `emo.sty` with the package implementation;
- (b) `emo.sty.ltxml` with the binding for [LaTeXML](#);
- (c) `emo.def` with the emoji table;
- (d) `emo-lingchi.ttf` with the two glyphs for `\lingchi`;
- (e) `emo-graphics` with the fallback PDF graphics.

TeX Live requires that each package's files have unique names. For that reason, the PDF graphics in the `emo-graphics` directory start with the `emo-` prefix as well.

When running on the LuaLaTeX engine, the `emo` package also uses the Noto color emoji (`NotoColorEmoji.ttf`) and Linux Libertine (`LinLibertine_R.otf`) fonts, with the latter used for rendering `\YHWH` only. Neither file is included with `emo`'s distribution, since both of them are distributed with major TeX distributions already. If they are not included with your LaTeX distribution, you can find them on CTAN. The `emo-lingchi.ttf` font distributed with `emo` is a two glyph subset of `NotoSerifTC-Regular.otf`, i.e., the traditional Chinese version of Noto serif.

## 2 Usage

As usual, you declare your document's dependency on `emo` with `\usepackage{emo}`. In addition to the unadorned form, `emo` takes up to two options:

**extra** Also define the `\lingchi` and `\YHWH` macros, which produce `凌遲` and `𠂇`.

**index** Create an emoji index tagged `emo` with the `.edx` extension for the raw index and the `.end` extension for the processed index. This option relies on the `index` package, generates the raw `.edx` file, but does not build or use the processed index.

### 2.1 One Main Macro

`\emo` An `\emo{\<emoji-name>}` invocation expands to the named emoji. For LuaLaTeX, it uses the Noto color emoji font. For all other engines, it uses PDF graphics. That way, `\emo{desert-island}` results in  and `\emo{parrot}` results in .

Since LaTeX tends to produce a lot of command line noise about underfull boxes and loaded fonts, it's a easy to miss meaningful warnings. For that reason, `\emo` expands to an attention-seeking error message upon undefined emoji names. For example, `\emo{boo}` produces `Bad \emo{boo}`.

#### 2.1.1 Naming Scheme

With a few exceptions, `emo`'s names for emoji are automatically derived from their Unicode names, with letters converted to lowercase, punctuation such as commas, colons, quotes, and parentheses stripped, and interword spaces replaced by dashes. Furthermore, instead of the rather verbose dark-skin-tone, medium-dark-skin-tone, etc modifiers, `emo` uses the more succinct darkest, darker, medium, lighter, and lightest.

For some names, `emo` goes further by hard-coding shorter names. Those names are listed in Table 1.

`Emo`'s `emo.def` contains the names and codepoints of all currently supported emoji in Unicode display order. `Emo`'s distribution also includes the `emoji-test.txt` file, which is part of [Unicode TR-51](#) and contains the names and codepoints of all *potentially supported* emoji, i.e., all emoji, also in Unicode display order. It further organizes emoji into groups and subgroups, with the current (sub)group being the one named on the closest line above the emoji that starts with # (sub)group:. As described in the next section, the group and subgroup names can be used during configuration for concisely naming a large number of emoji.

Table 1: Exceptional emoji names

Transformed Unicode Name	Emo Replacement Name
a-button-blood-type	a-button
ab-button-blood-type	ab-button
b-button-blood-type	b-button
o-button-blood-type	o-button
bust-in-silhouette	bust
busts-in-silhouette	busts
flag-european-union	eu
globe-showing-americas	globe-americas
globe-showing-asia-australia	globe-asia-australia
globe-showing-europe-africa	globe-africa-europe
hear-no-evil-monkey	hear-no-evil
index-pointing-at-the-viewer	index-pointing-at-viewer
index-pointing-at-the-viewer-darkest	index-pointing-at-viewer-darkest
index-pointing-at-the-viewer-darker	index-pointing-at-viewer-darker
index-pointing-at-the-viewer-medium	index-pointing-at-viewer-medium
index-pointing-at-the-viewer-lighter	index-pointing-at-viewer-lighter
index-pointing-at-the-viewer-lightest	index-pointing-at-viewer-lightest
keycap-*	keycap-star
keycap-#	keycap-hash
keycap-0	keycap-zero
keycap-1	keycap-one
keycap-2	keycap-two
keycap-3	keycap-three
keycap-4	keycap-four
keycap-5	keycap-five
keycap-6	keycap-six
keycap-7	keycap-seven
keycap-8	keycap-eight
keycap-9	keycap-nine
keycap-10	keycap-ten
magnifying-glass-tilted-left	loupe-left
magnifying-glass-tilted-right	loupe-right
palm-down-hand	palm-down
palm-down-hand-darkest	palm-down-darkest
palm-down-hand-darker	palm-down-darker
palm-down-hand-medium	palm-down-medium
palm-down-hand-lighter	palm-down-lighter
palm-down-hand-lightest	palm-down-lightest
palm-up-hand	palm-up
palm-up-hand-darkest	palm-up-darkest
palm-up-hand-darker	palm-up-darker
palm-up-hand-medium	palm-up-medium
palm-up-hand-lighter	palm-up-lighter
palm-up-hand-lightest	palm-up-lightest
rolling-on-the-floor-laughing	rofl
see-no-evil-monkey	see-no-evil
speak-no-evil-monkey	speak-no-evil

### 2.1.2 Inventory of Default Configuration

Emo's default configuration includes all emoji *but* those in the *people-and-body* Unicode group that modify gender, hair, or skin and those in the *country-flag* subgroup of the *flags* group other than the EU flag (which is included). As of Unicode 15.0, that's 1,415 out of 3,655 emoji. There are another 257 flags, but the majority of excluded emoji are variations on already included emoji depicting people or body parts. As described in the next section, enabling additional emoji is as easy as running a Python script with the emoji names as arguments.

With their names listed in Unicode display order and organized by Unicode group and subgroup, that's the following emoji.

<i>smileys-and-emotion</i>	
<hr/>	
<b>FACE-SMILING</b>	
grinning-face .....	😊
grinning-face-with-big-eyes .....	😍
grinning-face-with-smiling-eyes .....	😁
beaming-face-with-smiling-eyes .....	🤗
grinning-squinting-face .....	😆
grinning-face-with-sweat .....	😅
rofl .....	🤣
face-with-tears-of-joy .....	😂
slightly-smiling-face .....	☺️
upside-down-face .....	🙃
melting-face .....	😭
winking-face .....	😉
smiling-face-with-smiling-eyes .....	😊
smiling-face-with-halo .....	😇
<hr/>	
<b>FACE-AFFECTION</b>	
smiling-face-with-hearts .....	😍
smiling-face-with-heart-eyes .....	😍
<hr/>	
<b>FACE-HAND</b>	
smiling-face-with-open-hands .....	😊
face-with-hand-over-mouth .....	😘
face-with-open-eyes-and-hand-over-mouth .....	😘
face-with-peeking-eye .....	👀
<hr/>	
<b>FACE-SLEEPY</b>	
relieved-face .....	😌
pensive-face .....	😔
sleepy-face .....	😴
drooling-face .....	🤤
sleeping-face .....	😴

<hr/>	
FACE-UNWELL	astonished-face ..... 😲    clown-face ..... 🎭
face-with-medical-mask ..... 😷    face-with-thermometer ..... 😰	flushed-face ..... 🥱    ogre ..... 🦖
face-with-head-bandage ..... 😵    nauseated-face ..... 😳	pleading-face ..... 🥺    goblin ..... 🧟
face-vomiting ..... 🤢    sneezing-face ..... 😵	face-holding-back-tears ..... 😢    ghost ..... 🕷️
hot-face ..... 😵    cold-face ..... 😴	frowning-face-with-open-mouth ..... 😠    alien ..... 👽
woozy-face ..... 😵    face-with-crossed-out-eyes ..... 😵	anguished-face ..... 😰    alien-monster ..... 🦸
face-with-spiral-eyes ..... 😶    exploding-head ..... 😱	fearful-face ..... 😰    robot ..... 🤖
<hr/>	
FACE-HAT	anxious-face-with-sweat ..... 😰    CAT-FACE
cowboy-hat-face ..... 🤠    partying-face ..... 🎉	grinning-cat ..... 😃    grinning-cat-with-smiling-eyes ..... 😊
disguised-face ..... 😲    yawning-face ..... 😴	crying-face ..... 😢    cat-with-tears-of-joy ..... 😢
<hr/>	
FACE-GLASSES	loudly-crying-face ..... 😭    smiling-cat-with-heart-eyes ..... 😍
smiling-face-with-sunglasses ..... 😎    nerd-face ..... 😡	face-screaming-in-fear ..... 😱    cat-with-wry-smile ..... 😏
face-with-monocle.... 😎	confounded-face ..... 😐    kissing-cat ..... 😻
<hr/>	
FACE-CONCERNED	persevering-face .... 😔    weary-cat ..... 😓
confused-face ..... 😰    face-with-diagonal-mouth ..... 😰	disappointed-face.... 😔    crying-cat ..... 😢
worried-face..... 😰    slightly-frowning-face ..... 😰	downcast-face-with-sweat ..... 😔    pouting-cat..... 😻
frowning-face ..... 😰    face-with-open-mouth .....	weary-face ..... 😔    MONKEY-FACE
hushed-face..... 😰	yawning-face ..... 😴
<hr/>	
FACE-NEGATIVE	see-no-evil ..... 🙄
face-with-steam-from-nose ..... 😣	hear-no-evil ..... 🙄
enraged-face ..... 😡	speak-no-evil ..... 🙄
<hr/>	
FACE-COSTUME	HEART
love-letter ..... 💬	love-letter ..... 💬
heart-with-arrow ..... 💕	heart-with-ribbon.... 💕
angry-face-with-horns ..... 😡	sparkling-heart ..... 💫
skull ..... 💀	growing-heart ..... 💕
skull-and-crossbones ..... 💀	beating-heart ..... 💕
<hr/>	
pile-of-poo..... 💩	revolving-hearts .... 💕
<hr/>	
FACE-COSTUME	two-hearts ..... 💕
heart-decoration ..... 💋	heart-exclamation.... ❤
broken-heart ..... 💔	broken-heart ..... 💔
heart-on-fire ..... 💕	heart-on-fire ..... 💕

mending-heart .....	❤️	rightwards-hand .....	👉	HANDS
red-heart.....	❤️	leftwards-hand.....	👈	
pink-heart .....		palm-down.....	👉👈	
orange-heart.....	🧡	palm-up .....	👉👉	
yellow-heart.....	💛	leftwards-pushing- hand.....	👉👈👉	
green-heart.....	💚	rightwards-pushing- hand.....	👉👉👉	
blue-heart .....	💙			
light-blue-heart .....				
purple-heart.....	💜			
brown-heart.....	🤎			
black-heart.....	🖤			
grey-heart .....		ok-hand .....	👌	
white-heart.....	🤍	pinched-fingers .....	✌️	
		pinching-hand .....	✌️	
EMOTION		victory-hand.....	✌️	
		crossed-fingers .....	✌️	
kiss-mark.....	💋	hand-with-index- finger-and-thumb- crossed .....	👉👉👉	
hundred-points.....	💯	love-you-gesture .....	👉👉	
anger-symbol.....	💢	sign-of-the-horns....	👉👉	
collision.....	💥	call-me-hand.....	👉👉	
dizzy .....	💫			
sweat-droplets.....	💦			
dashing-away.....	💨			
hole .....	🕳️			
speech-balloon.....	💬			
eye-in-speech- bubble.....	👁️			
left-speech- bubble.....	🗨️			
right-anger- bubble.....	🗯️			
thought-balloon .....	💭			
zzz.....	💤			
<b>people-and-body</b>				
HAND-FINGERS-OPEN				
waving-hand.....	👋			
raised-back-of- hand .....	✋			
hand-with-fingers- splayed .....	🖐️			
raised-hand.....	✋			
vulcan-salute .....	🖖️			
		HAND-FINGERS-CLOSED		
thumbs-up.....	👍			
thumbs-down.....	👎			
raised-fist.....	✊			
oncoming-fist .....	✊			
left-facing-fist .....	✊			
right-facing-fist....	✊			

PERSON	
baby .....	
child .....	
boy.....	
girl .....	
person.....	
person-blond-hair....	
person-beard.....	
older-person.....	
PERSON-GESTURE	
person-frowning .....	
person-pouting.....	
person-gesturing-no .....	
person-gesturing-ok .....	
person-tipping-hand .....	
person-raising-hand .....	
deaf-person.....	
person-bowing .....	
person-facepalming.....	
person-shrugging .....	
PERSON-ROLE	
health-worker .....	
student .....	
teacher .....	
judge .....	
farmer.....	
cook .....	
mechanic .....	
factory-worker.....	
office-worker .....	
scientist.....	
technologist.....	
singer.....	
artist.....	
pilot .....	
astronaut.....	
firefighter.....	
police-officer.....	
detective.....	
guard .....	
ninja .....	
construction-worker.....	
person-with-crown....	
prince.....	
princess .....	
person-wearing-turban.....	
person-with-skullcap .....	
person-in-tuxedo .....	
person-with-veil .....	
pregnant-person.....	
breast-feeding.....	
person-feeding-baby .....	
PERSON-FANTASY	
baby-angel .....	
santa-claus.....	
mrs-claus.....	
mx-claus .....	
superhero.....	
supervillain.....	
mage .....	
fairy .....	
vampire .....	
merperson.....	
merman.....	
mermaid .....	
elf.....	
genie .....	
zombie.....	
troll .....	
PERSON-ACTIVITY	
person-getting-massage .....	
person-getting-haircut .....	
person-walking.....	
person-standing .....	
person-kneeling .....	
person-with-white-cane .....	
person-in-motorized-wheelchair .....	
person-in-manual-wheelchair .....	
person-running.....	
person-in-suit-levitating .....	
people-with-bunny-ears .....	
person-in-steamy-room .....	
person-climbing .....	
PERSON-SPORT	
person-fencing.....	
horse-racing .....	
skier .....	
snowboarder.....	
person-golfing.....	
person-surfing.....	
person-rowing-boat .....	
person-swimming .....	
person-bouncing-ball .....	
person-lifting-weights .....	
person-biking .....	
person-mountain-biking .....	
person-cartwheeling .....	
people-wrestling .....	
person-playing-water-polо .....	
person-playing-handball .....	
person-juggling .....	

<hr/>	
PERSON-RESTING	lion .....  sloth .....
person-in-lotus-position .....	tiger-face .....  otter .....
person-taking-bath .....	tiger .....  skunk .....
person-in-bed .....	leopard .....  kangaroo .....
<hr/>	
FAMILY	badger .....  paw-prints .....
people-holding-hands .....	donkey .....
kiss .....	horse .....
couple-with-heart .....	unicorn .....
family .....	zebra .....
<hr/>	
PERSON-SYMBOL	deer .....  rooster .....
speaking-head .....	bison .....  hatching-chick .....
bust .....	cow-face .....  baby-chick .....
busts .....	ox .....  front-facing-baby-chick .....
people-hugging .....	water-buffalo .....  cow .....
footprints .....	pig-face .....  bird .....
<hr/>	
<i><b>animals-and-nature</b></i>	
<hr/>	
ANIMAL-MAMMAL	pig .....  penguin .....
monkey-face .....	boar .....  dove .....
monkey .....	pig-nose .....  eagle .....
gorilla .....	ram .....  duck .....
orangutan .....	ewe .....  swan .....
dog-face .....	goat .....  owl .....
dog .....	camel .....  dodo .....
guide-dog .....	two-hump-camel .....  feather .....
service-dog .....	llama .....  flamingo .....
poodle .....	giraffe .....  peacock .....
wolf .....	elephant .....  parrot .....
fox .....	mammoth .....  wing .....
raccoon .....	hippopotamus .....  black-bird .....
cat-face .....	mouse-face .....  goose .....
cat .....	mouse .....
black-cat .....	rat .....
<hr/>	
ANIMAL-AMPHIBIAN	
<hr/>	
ANIMAL-REPTILE	
<hr/>	
9	

sauropod .....		rose .....		cherries .....			
t-rex .....		wilted-flower .....		strawberry .....			
<hr/>							
ANIMAL-MARINE		hibiscus .....		blueberries .....			
spouting-whale.....		sunflower .....		kiwi-fruit .....			
whale .....		blossom .....		tomato .....			
dolphin .....		tulip .....		olive .....			
seal .....		hyacinth .....		coconut .....			
<hr/>							
PLANT-OTHER		<hr/>					
fish .....		seedling .....		avocado .....			
tropical-fish .....		potted-plant .....		eggplant .....			
blowfish .....		evergreen-tree .....		potato .....			
shark .....		deciduous-tree .....		carrot .....			
octopus .....		palm-tree .....		ear-of-corn .....			
spiral-shell .....		cactus .....		hot-pepper .....			
coral .....		sheaf-of-rice .....		bell-pepper .....			
jellyfish .....		herb .....		cucumber .....			
<hr/>							
ANIMAL-BUG		shamrock .....		leafy-green .....			
snail .....		four-leaf-clover .....		broccoli .....			
butterfly .....		maple-leaf .....		garlic .....			
bug .....		fallen-leaf .....		onion .....			
ant .....		leaf-fluttering-in-wind .....		peanuts .....			
honeybee .....		empty-nest .....		beans .....			
beetle .....		nest-with-eggs .....		chestnut .....			
lady-beetle .....		mushroom .....		ginger-root .....			
cricket .....				pea-pod .....			
cockroach .....		<hr/>					
spider .....		<hr/>					
spider-web .....		FOOD-FRUIT		FOOD-PREPARED			
scorpion .....		grapes .....		bread .....			
mosquito .....		melon .....		croissant .....			
fly .....		watermelon .....		baguette-bread .....			
worm .....		tangerine .....		flatbread .....			
microbe .....		lemon .....		pretzel .....			
<hr/>							
PLANT-FLOWER		banana .....		bagel .....			
bouquet .....		pineapple .....		pancakes .....			
cherry-blossom .....		mango .....		waffle .....			
white-flower .....		red-apple .....		cheese-wedge .....			
lotus .....		green-apple .....		meat-on-bone .....			
rosette .....		pear .....		poultry-leg .....			
		peach .....		cut-of-meat .....			
				bacon .....			
				hamburger .....			

french-fries .....		FOOD-MARINE	
pizza .....			cup-with-straw .....
hot-dog .....		crab .....	bubble-tea .....
sandwich .....		lobster .....	beverage-box .....
taco .....		shrimp .....	mate .....
burrito .....		squid .....	ice .....
tamale .....		oyster .....	
stuffed-flatbread....			DISHWARE
falafel .....			
egg.....		FOOD-SWEET	chopsticks .....
cooking .....			fork-and-knife-with-plate .....
shallow-pan-of-food .....		soft-ice-cream.....	fork-and-knife.....
pot-of-food.....		shaved-ice .....	spoon .....
fondue.....		ice-cream.....	kitchen-knife .....
bowl-with-spoon .....		doughnut .....	jar .....
green-salad.....		cookie.....	amphora .....
popcorn .....		birthday-cake .....	
butter.....		shortcake.....	
salt.....		cupcake .....	
canned-food.....		pie.....	
FOOD-ASIAN		chocolate-bar .....	
		candy .....	
bento-box .....		lollipop .....	
rice-cracker .....		custard .....	
rice-ball.....		honey-pot .....	
cooked-rice.....		DRINK	
curry-rice .....		baby-bottle.....	
steaming-bowl .....		glass-of-milk .....	
spaghetti.....		hot-beverage .....	
roasted-sweet-potato.....		teapot.....	
oden.....		teacup-without-handle.....	
sushi .....		sake .....	
fried-shrimp.....		bottle-with-popping-cork .....	
fish-cake-with-swirl .....		wine-glass .....	
moon-cake.....		cocktail-glass .....	
dango .....		tropical-drink .....	
dumpling .....		beer-mug .....	
fortune-cookie.....		clinking-beer-mugs .....	
takeout-box.....		clinking-glasses .....	
		tumbler-glass .....	

PLACE-BUILDING	
stadium .....	
classical-building .....	
building-construction.....	
brick .....	
rock .....	
wood .....	
hut.....	
houses.....	
derelict-house.....	
house .....	
house-with-garden....	
office-building .....	
japanese-post-office.....	
post-office.....	
hospital .....	
bank .....	
hotel .....	
love-hotel .....	
convenience-store....	
school.....	
department-store .....	
factory .....	
japanese-castle .....	
castle.....	
wedding .....	
tokyo-tower.....	
statue-of-liberty....	
TRANSPORT-GROUND	
tent .....	
foggy .....	
night-with-stars .....	
cityscape.....	
sunrise-over-mountains.....	
sunrise .....	
cityscape-at-dusk....	
sunset.....	
bridge-at-night.....	
hot-springs.....	
carousel-horse.....	
playground-slide .....	
ferris-wheel.....	
roller-coaster.....	
barber-pole.....	
circus-tent.....	
TRANSPORT-WATER	
oncoming-automobile .....	
sport-utility-vehicle .....	
pickup-truck .....	
delivery-truck.....	
articulated-lorry....	
tractor .....	
racing-car .....	
motorcycle .....	
motor-scooter .....	
manual-wheelchair....	
motorized-wheelchair .....	
auto-rickshaw .....	
bicycle .....	
kick-scooter .....	
skateboard .....	
roller-skate .....	
bus-stop .....	
motorway .....	
railway-track .....	
oil-drum .....	
fuel-pump.....	
wheel .....	
police-car-light .....	
horizontal-traffic-light .....	
vertical-traffic-light .....	
stop-sign.....	
construction.....	
PLACE-RELIGIOUS	
church.....	
mosque.....	
hindu-temple.....	
synagogue.....	
shinto-shrine .....	
kaaba .....	
PLACE-OTHER	
fountain .....	
automobile .....	
ship.....	

<b>TRANSPORT-AIR</b>	airplane .....		five-thirty.....		sun-behind-cloud .....	
	small-airplane.....		six-oclock .....		cloud-with-lightning-and-rain .....	
	airplane-departure.....		six-thirty .....		sun-behind-small-cloud .....	
	airplane-arrival .....		seven-oclock.....		sun-behind-large-cloud .....	
	parachute.....		seven-thirty .....		sun-behind-rain-cloud .....	
	seat .....		eight-oclock.....		cloud-with-rain .....	
	helicopter .....		eight-thirty .....		cloud-with-snow .....	
	suspension-railway .....		nine-oclock.....		cloud-with-lightning .....	
	mountain-cableway.....		nine-thirty .....		tornado .....	
	aerial-tramway.....		ten-oclock .....		fog.....	
	satellite.....		ten-thirty .....		wind-face.....	
	rocket.....		eleven-oclock .....		cyclone .....	
	flying-saucer .....		eleven-thirty .....		rainbow .....	
<b>HOTEL</b>	bellhop-bell .....		<b>SKY-AND-WEATHER</b>		closed-umbrella .....	
	luggage .....		new-moon .....		umbrella .....	
<b>TIME</b>	hourglass-done.....		waxing-crescent-moon .....		umbrella-with-rain-drops .....	
	hourglass-not-done .....		first-quarter-moon .....		umbrella-on-ground .....	
	watch .....		waxing-gibbous-moon .....		high-voltage .....	
	alarm-clock.....		full-moon.....		snowflake.....	
	stopwatch.....		waning-gibbous-moon .....		snowman .....	
	timer-clock.....		last-quarter-moon.....		snowman-without-snow .....	
	mantelpiece-clock....		waning-crescent-moon .....		comet .....	
	twelve-o'clock .....		crescent-moon .....		fire .....	
	twelve-thirty .....		new-moon-face .....		droplet .....	
	one-o'clock .....		first-quarter-moon-face .....		water-wave .....	
	one-thirty .....		last-quarter-moon-face .....			
	two-o'clock .....		thermometer.....			
	two-thirty .....		sun.....		<b>activities</b>	
	three-o'clock.....		full-moon-face.....		<b>EVENT</b>	
	three-thirty.....		sun-with-face .....		jack-o-lantern.....	
	four-o'clock.....		ringed-planet .....		christmas-tree.....	
	four-thirty.....		star .....		fireworks.....	
	five-o'clock.....		glowing-star .....		sparkler .....	
			shooting-star .....		firecracker.....	
			milky-way .....		sparkles .....	
			cloud .....		balloon .....	

## *activities*

EVENT	
jack-o-lantern.....	
christmas-tree.....	
fireworks.....	
sparkler .....	
firecracker.....	
sparkles .....	
balloon .....	

party-popper .....		martial-arts-		artist-palette .....																																																																																																											
confetti-ball .....		uniform .....		thread .....																																																																																																											
tanabata-tree .....		goal-net .....		sewing-needle .....																																																																																																											
pine-decoration .....		flag-in-hole .....		yarn .....																																																																																																											
japanese-dolls .....		ice-skate .....		knot .....																																																																																																											
carp-streamer .....		fishing-pole .....		<b>objects</b>																																																																																																											
wind-chime .....		diving-mask .....																																																																																																													
moon-viewing-		running-shirt .....		<hr/>																																																																																																											
ceremony .....		skis .....		CLOTHING																																																																																																											
red-envelope .....		sled .....																																																																																																													
ribbon .....		curling-stone .....																																																																																																													
wrapped-gift .....		<hr/>																																																																																																													
reminder-ribbon .....		<hr/>																																																																																																													
admission-tickets....		<hr/>																																																																																																													
ticket .....		<hr/>																																																																																																													
<hr/>																																																																																																															
AWARD-MEDAL																																																																																																															
military-medal .....		bullseye .....		glasses .....																																																																																																											
trophy .....		yo-yo .....		sunglasses .....																																																																																																											
sports-medal .....		kite .....		goggles .....																																																																																																											
1st-place-medal .....		water-pistol .....		lab-coat .....																																																																																																											
2nd-place-medal .....		pool-8-ball .....		safety-vest .....																																																																																																											
3rd-place-medal .....		crystal-ball .....		necktie .....																																																																																																											
<hr/>																																																																																																															
SPORT																																																																																																															
soccer-ball .....		piñata .....		t-shirt .....																																																																																																											
baseball .....		mirror-ball .....		jeans .....																																																																																																											
softball .....		nesting-dolls .....		scarf .....																																																																																																											
basketball .....		spade-suit .....		gloves .....																																																																																																											
volleyball .....		heart-suit .....		coat .....																																																																																																											
american-football....		diamond-suit .....		socks .....																																																																																																											
rugby-football .....		club-suit .....		dress .....																																																																																																											
tennis .....		chess-pawn .....		kimono .....																																																																																																											
flying-disc .....		joker .....		sari .....																																																																																																											
bowling .....		mahjong-red-		one-piece-																																																																																																											
cricket-game .....		dragon .....		swimsuit .....																																																																																																											
field-hockey .....		flower-playing-		ice-hockey .....		cards .....		briefs .....		lacrosse .....		<hr/>		shorts .....		ping-pong .....		<hr/>		bikini .....		badminton .....		<hr/>		womans-clothes .....		boxing-glove .....		<hr/>		folding-hand-fan .....				<hr/>		purse .....				<hr/>		handbag .....				<hr/>		clutch-bag .....				<hr/>		shopping-bags .....				<hr/>		backpack .....				<hr/>		thong-sandal .....				<hr/>		mans-shoe .....				<hr/>		running-shoe .....				<hr/>		hiking-boot .....				<hr/>		flat-shoe .....				<hr/>		high-heeled-shoe .....				<hr/>		womans-sandal .....				<hr/>		ballet-shoes .....	
ice-hockey .....		cards .....		briefs .....																																																																																																											
lacrosse .....		<hr/>		shorts .....																																																																																																											
ping-pong .....		<hr/>		bikini .....																																																																																																											
badminton .....		<hr/>		womans-clothes .....																																																																																																											
boxing-glove .....		<hr/>		folding-hand-fan .....																																																																																																											
		<hr/>		purse .....																																																																																																											
		<hr/>		handbag .....																																																																																																											
		<hr/>		clutch-bag .....																																																																																																											
		<hr/>		shopping-bags .....																																																																																																											
		<hr/>		backpack .....																																																																																																											
		<hr/>		thong-sandal .....																																																																																																											
		<hr/>		mans-shoe .....																																																																																																											
		<hr/>		running-shoe .....																																																																																																											
		<hr/>		hiking-boot .....																																																																																																											
		<hr/>		flat-shoe .....																																																																																																											
		<hr/>		high-heeled-shoe .....																																																																																																											
		<hr/>		womans-sandal .....																																																																																																											
		<hr/>		ballet-shoes .....																																																																																																											

womans-boot.....		MUSICAL-INSTRUMENT	LIGHT-AND-VIDEO		
hair-pick.....					
crown .....		saxophone.....			
womans-hat .....		accordion.....			
top-hat .....		guitar.....			
graduation-cap.....		musical-keyboard .....			
billed-cap .....		trumpet .....			
military-helmet .....		violin.....			
rescue-workers-helmet.....		banjo .....			
prayer-beads.....		drum.....			
lipstick .....		long-drum.....			
ring .....		maracas .....			
gem-stone.....		flute .....			
<hr/>					
PHONE					
<hr/>					
SOUND		mobile-phone .....			
muted-speaker .....		mobile-phone-with-arrow .....			
speaker-low-volume.....		telephone.....			
speaker-medium-volume.....		telephone-receiver .....			
speaker-high-volume.....		pager .....			
loudspeaker.....		fax-machine.....			
megaphone.....		<hr/>			
postal-horn.....		COMPUTER			
bell .....		battery .....			
bell-with-slash.....		low-battery.....			
<hr/>					
MUSIC		electric-plug .....			
musical-score .....		laptop.....			
musical-note.....		desktop-computer .....			
musical-notes .....		printer .....			
studio-microphone....		keyboard .....			
level-slider.....		computer-mouse.....			
control-knobs .....		trackball.....			
microphone .....		computer-disk .....			
headphone.....		floppy-disk.....			
radio .....		optical-disk.....			
<hr/>					
MONEY					
<hr/>					
		money-bag .....			
		coin.....			

yen-banknote .....	¥	file-folder.....	📁	bow-and-arrow .....	🏹
dollar-banknote .....	\$	open-file-folder .....	📁	shield.....	🛡
euro-banknote .....	€	card-index-.....	📁	carpentry-saw .....	🪚
pound-banknote.....	£	dividers .....	📁	wrench.....	🔧
money-with-wings .....	💰	calendar .....	📅	screwdriver.....	🔧
credit-card.....	💳	tear-off-calendar....	📅	nut-and-bolt .....	🔩
receipt .....	📄	spiral-notepad.....	📝	gear .....	⚙️
chart-increasing-with-yen.....	💹	spiral-calendar .....	📅	clamp .....	_CLAMP
MAIL		card-index .....	📁	balance-scale .....	⚖️
envelope .....	✉️	chart-decreasing .....	📉	white-cane .....	🦵
e-mail.....	✉️	bar-chart.....	📊	link.....	🔗
incoming-envelope....	✉️	clipboard.....	📋	chains.....	⛓
envelope-with-arrow .....	✉️	pushpin .....	📌	hook.....	훅
outbox-tray.....	📦	round-pushpin .....	📌	toolbox .....	.toolbox
inbox-tray .....	📦	paperclip.....	📎	magnet.....	🧲
package .....	📦	linked-paperclips....	📎	ladder.....	🪜
closed-mailbox-with-raised-flag.....	📫	straight-ruler.....	📏		
closed-mailbox-with-lowered-flag.....	📫	triangular-ruler ....	📐		
open-mailbox-with-raised-flag.....	📫	scissors .....	✂️		
open-mailbox-with-lowered-flag.....	📫	card-file-box .....	🗄️		
postbox .....	📮	file-cabinet.....	🗄️		
ballot-box-with-ballot.....	🗳️	wastebasket.....	🗑️		
WRITING				SCIENCE	
pencil.....	-pencil			alembic .....	/distillation
black-nib.....	-nib			test-tube.....	🧪
fountain-pen.....	-pen			petri-dish .....	玱玱
pen.....	-pen			dna.....	DNA
paintbrush .....	-brush			microscope .....	🔬
crayon.....	-crayon			telescope.....	🔭
memo .....	-memo			satellite-antenna....	📡
OFFICE					
briefcase.....	💼	TOOL			
		hammer .....	🔨		
		axe.....	斧		
		pick .....	⛏️		
		hammer-and-pick .....	🔨⛏️		
		hammer-and-wrench....	🔨🔧		
		dagger .....	🗡️		
		crossed-swords.....	⚔️		
		bomb .....	💣		
		boomerang.....	🎯	MEDICAL	
				syringe .....	💉
				drop-of-blood .....	滴血
				pill.....	💊
				adhesive-bandage .....	绷带
				crutch.....	拄杖
				stethoscope.....	听诊器
				x-ray .....	透视

<b>symbols</b>	
<b>HOUSEHOLD</b>	
door .....	
elevator .....	
mirror .....	
window.....	
bed.....	
couch-and-lamp.....	
chair .....	
toilet.....	
plunger .....	
shower.....	
bathtub .....	
mouse-trap .....	
razor .....	
lotion-bottle .....	
safety-pin .....	
broom .....	
basket.....	
roll-of-paper .....	
bucket.....	
soap.....	
bubbles .....	
toothbrush .....	
sponge.....	
fire-extinguisher....	
shopping-cart .....	
<b>OTHER-OBJECT</b>	
cigarette.....	
coffin.....	
headstone.....	
funeral-urn.....	
nazar-amulet.....	
hamsa .....	
moai .....	
placard .....	
identification-card .....	
<b>TRANSPORT-SIGN</b>	
atm-sign .....	
litter-in-bin-sign.....	
potable-water .....	
wheelchair-symbol....	
mens-room.....	
womens-room.....	
restroom .....	
baby-symbol.....	
water-closet.....	
passport-control ....	
customs .....	
baggage-claim .....	
left-luggage .....	
<b>WARNING</b>	
warning .....	
children-crossing....	
no-entry .....	
prohibited .....	
no-bicycles.....	
no-smoking .....	
no-littering.....	
non-potable-water....	
no-peDESTrians.....	
no-mobile-phones .....	
no-one-under-eighteen .....	
radioactive.....	
biohazard.....	
<b>ARROW</b>	
up-arrow .....	
up-right-arrow.....	
right-arrow.....	
down-right-arrow ....	
down-arrow .....	
down-left-arrow .....	
<b>RELIGION</b>	
place-of-worship .....	
atom-symbol.....	
om .....	
star-of-david .....	
wheel-of-dharma .....	
yin-yang .....	
latin-cross.....	
orthodox-cross.....	
star-and-crescent....	
peace-symbol .....	
menorah .....	
dotted-six-pointed-star .....	
khanda.....	
<b>ZODIAC</b>	
aries .....	
taurus.....	
gemini.....	
cancer.....	

leo.....	♌	GENDER	japanese-symbol-for-
virgo.....	♍		beginner .....
libra.....	♎		hollow-red-circle....
scorpio.....	♏	female-sign.....	check-mark-button....
sagittarius.....	♐	male-sign.....	check-box-with-
capricorn.....	♑	transgender-	check.....
aquarius.....	♒	symbol.....	check-mark.....
pisces.....	♓		cross-mark.....
ophiuchus.....	♉	MATH	cross-mark-button....
			curly-loop.....
			double-curly-loop....
			part-alternation-
			mark.....
shuffle-tracks-	X		eight-spoked-
button.....	X		asterisk .....
repeat-button.....	⟳		eight-pointed-
repeat-single-	⟳		star.....
button.....	⟳	PUNCTUATION	sparkle .....
play-button.....	▶		copyright.....
fast-forward-	▶		registered.....
button.....	▶		trade-mark .....
next-track-button....	▶▶	double-exclamation-	TM
play-or-pause-	▶▶	mark .....	
button.....	▶▶	exclamation-question-	
reverse-button.....	◀	mark .....	
fast-reverse-	◀	red-question-mark....	
button.....	◀	white-question-	
last-track-button....	◀◀	mark .....	
upwards-button.....	▲	white-exclamation-	
fast-up-button.....	▲	mark .....	
downwards-button .....	▼	red-exclamation-	
fast-down-button ....	▼	mark .....	
pause-button.....	⏸	wavy-dash.....	
stop-button.....	⏹	CURRENCY	
record-button .....	⏺		keycap-hash.....
eject-button.....	⏏		keycap-star.....
cinema.....	🎦		keycap-zero.....
dim-button .....	☀		keycap-one .....
bright-button .....	☀	OTHER-SYMBOL	keycap-two .....
antenna-bars.....	📶		keycap-three .....
wireless .....	📶	medical-symbol.....	keycap-four .....
vibration-mode.....	📱	recycling-symbol .....	keycap-five .....
mobile-phone-off .....	📴	fleur-de-lis.....	keycap-six .....
		trident-emblem.....	keycap-seven .....
		name-badge .....	keycap-eight .....
			keycap-nine .....
			keycap-ten .....
			ALPHANUM
			input-latin-
			uppercase.....
			input-latin-
			lowercase.....
			input-numbers .....



## 2.2 Two Optional Macros

\lingchi The \lingchi and \YHWH macros take no arguments and produce 凌遲 and הָהָ. They \YHWH are only available if emo is used with the extra option. The former renders the Chinese term for “death by a thousand cuts.” While originally an execution method, the term applies to surprisingly many software systems as well. The latter produces the Tetragrammaton, the Hebrew name for God. Observant Jews never utter what’s written, not even in their thoughts, substituting Adonai (“My Lord”), Elohim (“God”), or HaShem (“The Name”) instead. In my mind, that nicely mirrors the very incomprehensibility of הָהָ. Both macros preserve a subsequent space as space, no backslash needed.

## 2.3 Conversion to HTML

Emo supports conversion to HTML with either [LaTeXML](#) or [TeX4ht](#). LaTeXML support is implemented by a separate “binding” against LaTeXML’s Perl API. I chronicled my exploration of suitable options leading to that less than ideal choice in a [GitHub issue](#). TeX4ht support is implemented by the emo package itself. It requires processing with LuaLaTeX e.g., by passing -l or --lua to the make4ht tool.

# 3 Configuration

Emo’s implementation is actually split over two files: `emo.sty` is extracted from `emo.dtx` and defines the substance of the package, its options, its helper macros, and the user-visible \emo, \lingchi, and \YHWH macros. Currently supported emoji are defined by the emoji table in the second file, `emo.def`. For every supported emoji, the file contains a command `\emo@emoji@(emoji-name)` with the emoji’s codepoints as value.

Configuration automates the regeneration of the emoji table for arbitrary numbers of emoji. `config/emo.py` is the script and `config/emoji-test.txt` is the list of all emoji from the Unicode standard.

## 3.1 Running the Configuration Script

To update emo’s configuration, invoke the `config/emo.py` script:

```
$ python3 config/emo.py <selector> <selector> ...
```

Each selector may be:

- The literal ALL (case-sensitive) for *all* emoji.
- Name of a group in `emoji-test.txt` lowercased and with spaces replaced by dashes and ampersand & replaced by an and; e.g., `travel-and-places`.
- Name of a group, a double colon ::, and name of a subgroup, again lowercased and with spaces replaced by dashes and & by an and; e.g., `travel-and-places::placegeographic`.
- The name of an emoji; e.g., `desert-island`.

For conjunctive group names, such as “Smileys & Emotion” (`emoji-test.txt`) or “smileys-and-emotion” (`emo.py`), the configuration script also accepts either of the two nouns as a shortcut, e.g., “smileys” or “emotion.”

For data safety, `emo.py` does not overwrite PDF graphics and hence can only *add* emoji to the configuration. To remove emoji, simply remove their PDF graphics from `emo-graphics` and then run `emo.py` without selector arguments, which updates the emoji table accordingly.

`emo.py` effectively treats `emoji-test.txt` as registry of all emoji and the filenames of PDF graphics in `emo-graphics` as `emo`'s current inventory. For all emoji named by selector arguments but not in the inventory, `emo.py` converts the SVG source graphic from the Noto color emoji sources to a PDF file and deletes the `/Page /Group` object from the the PDF again, since that object trips up `pdflatex`. And yeah, `emo.py` automatically downloads the Noto color emoji sources if necessary.

## 4 Copyright and Licensing

Since `emo`'s distribution includes not only LaTeX code but also a substantial Python script, Unicode data about emoji, as well as graphics and fonts derived from Google's Noto project, a number of different licenses apply. All of them are [OSI approved](#) and non-copyleft:

- This package's LaTeX and also Perl code extracted from `emo.dtx` is © Copyright 2023 by Robert Grimm and has been released under the [LPPL v1.3c](#) or later.
- The `config/emo.py` script also is © Copyright 2023 by Robert Grimm but has been released under the [Apache 2.0 license](#).
- The `[config/emoji-test.txt]` configuration file is a data file from [Unicode TR-51](#) and hence subject to the [Unicode License](#).
- The `emo-lingchi.ttf` font is a two-glyph subset of the traditional Chinese version of Google's [Noto serif](#) and hence subject to the [SIL Open Font License v1.1](#).
- The PDF graphics in the `emo-graphics` directory are derived from the sources for [Noto's color emoji](#) and hence subject to the Apache 2.0 license.

## 5 Implementation

Let's get started on `emo`'s implementation:

<sup>1</sup> `(*package)`

Except, that implementation started near the top of this `emo.dtx` file, well before the documentation's preamble. For completeness, here is the package declaration again:

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{emo}
[2023/99/99 v1.0 emo·ji for all (LaTeX engines)]
```

Unfortunately, `emo`'s version number appears in triplicate in this file, which makes cutting a release a little more tricky than it should be. Alas, the above, faux package declaration is not one of those repeat offenders. Really.

## 5.1 Package Options

\ifEmojiExtra Define a conditional flag for each package option.

\ifemo@index

\ifemo@debug 2 \newif\ifEmojiExtra

3 \newif\ifemo@index

4 \newif\ifemo@debug

Wire each conditional to the package option and then process them all.

5 \DeclareOption{extra}{\EmojiExtratrue}

6 \DeclareOption{index}{\emo@indextrue}

7 \DeclareOption{debug}{\emo@debugtrue}

8 \ProcessOptions\relax

## 5.2 Package Dependencies and Backend Selection

Require `inputenc` to declare this file's character encoding as UTF-8. XeTeX and LuaTeX already use that encoding by default and hence this line is redundant. But pdfTeX originates from darker, pre-Unicode times and thus needs to be told. Though thankfully, it does support the encoding by now.

9 \RequirePackage[utf8]{inputenc}

\ifemo@use@unicode Define a conditional flag for each major backend feature. They are *not* orthogonal:

\ifemo@use@font and \ifemo@use@pdf are mutually exclusive and determine \ifemo@use@pdf whether `emo` generates Unicode text or PDF graphics. For \ifemo@use@font to be enabled, \ifemo@use@unicode must already be enabled as well.

10 \newif\ifemo@use@unicode

11 \newif\ifemo@use@font

12 \newif\ifemo@use@pdf

Inspect the runtime environment and accordingly set the just defined backend flags. \HCode is defined by TeX4ht. Since correctly converting to HTML with TeX4ht requires LuaTeX, print a helpful error message for other engines.

13 \RequirePackage{iftex}

14 \ifdefinable\HCode

15 \ifluatex\else

16 \PackageError{emo}{%

17     LuaTeX required for TeX4ht converting LaTeX to HTML.\MessageBreak

18     You may also see ``Missing character'' messages or\MessageBreak

19     `Unicode character ... not set up'' errors.\MessageBreak

20     To fix, please pass the -l or --lua option to make4ht tool}{%

21     Run TeX4ht with LuaTeX by passing -l or --lua option to make4ht}

22 \fi

23 \emo@use@unicodetrue

24 \else

25 \ifluatex

26 \emo@use@unicodetrue

27 \emo@use@fonttrue

```

28 \else
29 \emo@use@pdftrue
30 \fi
31 \fi

```

We need the backend flags to decide whether to load `fontspec` or not. We do that before requiring any other packages because that way the packages related to encoding and fonts are loaded first (together with `iftex`).

```

32 \ifemo@use@font
33 \RequirePackage{fontspec}
34 \fi

```

Require `xcolor` for formatting highly visible error messages within the generated document. Always including another package that is only used when there are errors is not ideal. But I couldn't get on-demand package loading to work. So we have to eagerly require the package.

```
35 \RequirePackage{xcolor}
```

The two remaining requirements depends on a package option and on the backend, respectively.

```

36 \ifemo@index
37 \RequirePackage{index}
38 \fi
39 \ifemo@use@pdf
40 \RequirePackage{graphicx}
41 \fi

```

### 5.3 The Emoji Table

The emoji table contains an entry for every emoji supported in the current configuration. Each such entry defines a macro without arguments, which has the emoji's name prepended with `\emo@emoji@` as its name and expands to the emoji's fully qualified Unicode codepoints. The macro for `keycap-hash` is an exception: Its first codepoint is the plain hash `#` character, which trips up TeX's parser. Hence the macro for `keycap-hash` expands to the `\char`ed` codepoints. If the extra option is active, the emoji table also contains entries for `\lingchi` and `\YHWH`.

Before version 1.0, `emo.def` contained the macro definitions for the emoji table, with definitions ordered by emojis' codepoints. The lack of meaningful hierarchy and order made `emo.def` unusable for generating documentation, notably an inventory of currently supported emoji. To make `emo.def` more useful, version 1.0 introduced a higher-level file format based on the five macros described next.

`\EmojiBeginGroup` The file format for `emo.def` relies on `\DefineEmoji` to associate emoji names with their `\EmojiBeginSubgroup` Unicode codepoints, `\EmojiBeginGroup` and `\EmojiEndGroup` to organize emoji into `\DefineEmoji` groups, and on `\EmojiBeginSubgroup` and `\EmojiEndSubgroup` to organize emoji into `\EmojiEndSubgroup` subgroups. All definitions with `\DefineEmoji` are listed in Unicode display order and `\EmojiEndGroup` grouped in their Unicode group and subgroup. Subgroups are properly nested inside groups. The group and subgroup for `lingchi` and `YHWH` are both called `extra`.

`\DefineEmoji{<name>}{<codepoints>}` takes an emoji's name and its Unicode codepoints as arguments—except for `keycap`-hash, which uses has the `\char`ed` codepoints as value because TeX does not handle hash characters in the input.

`\EmojiBeginGroup{<group>}` and `\EmojiEndGroup{<group>}` take a group name as their only argument. In contrast, `\EmojiBeginSubgroup{<group>}{<subgroup>}` and `\EmojiEndSubgroup{<group>}{<subgroup>}` take both the group and subgroup names as their two arguments.

If the `extra` option is enabled, `\lingchi` and `\YHWH` are treated just like emoji, and their names and Unicode codepoints (sans any group, font selection, or text direction) are included with the emoji table.

The conversion from the abstract file format for `emo.def` to the concrete emoji table is simple enough, especially since four macros remain no-ops.

```
42 \def\EmojiBeginGroup#1{}
43 \def\EmojiBeginSubgroup#1#2{}
44 \def\DefineEmoji#1#2{%
45     \expandafter\def\csname emoji@#1\endcsname{#2}}
46 \def\EmojiEndSubgroup#1#2{}
47 \def\EmojiEndGroup#1{}
```

With that, we are ready to read `emo.def`:

```
48 \input{emo.def}
```

The emoji table is automatically generated with the `config/emo.py` Python script. When invoked without arguments, that script simply recreates the emoji table based on the emoji present in the `emo-graphics` directory. If invoked with arguments naming Unicode emoji groups, groups and subgroups, or individual emoji, the script converts SVG graphics from the Noto emoji font sources to PDF graphics compatible with LaTeX and then rebuilds the inventory in `emo.def`. If necessary, the Python code downloads the Noto emoji font sources.

## 5.4 Internal Macros

`\emo@error@fg` Define two colors and a macro `\emo@error{<emoji-name>}` function that uses the two `\emo@error@bg` colors for formatting an attention-grabbing error message. If you use an invalid emoji `\emo@error` name and overlook the warning in the console, you *will* notice the error message in the document thusly formatted.

```
49 \definecolor{emo@error@fg}{rgb}{1,1,1}
50 \definecolor{emo@error@bg}{rgb}{.6824,.0863,.0863}
51 \def\emo@error#1{%
52     \colorbox{emo@error@bg}{%
53         \textcolor{emo@error@fg}{%
54             \textsf{Bad} \texttt{\char`\\"emo\{#1\}}}}}
```

`\emo@ifdef` The `\emo@ifdef{<name>}{<code>}` macro validate the emoji name given as first argument. If the name is invalid, it expands to an error message. If the name is valid, it

executes the code given as second argument. The implementation critically relies on the emoji table being accurate.

```
55 \def\emo@ifdef#1#2{%
56   \ifcsname emoji@emoji#1\endcsname#2\else%
57     \PackageWarning{emo}{Unknown emoji name in `\\string\\emoji{#1}'}%
58     \emo@error{#1}%
59   \fi}
```

## 5.5 User Macros and Hooks

### 5.5.1 Background: Variability vs Code Duplication

Emo's implementation changed considerably between versions 0.1 and 1.0:

For **versions 0.1 and 0.2**, I focused on getting emo to work across pdfTeX, XeTeX, and LuaTeX as well as to adhere to the (naming) conventions of the TeX ecosystem. My initial strategy for managing the variability of package options and backends was to define the same internal and user macros several times each to account for just that variability. Hence, emo included two versions for the internal `\emo@index` macro, namely one version to emit an index entry and one version to do nothing. It also included two versions each for `\emo`, `\lingchi`, and `\YHWH`, namely one version to emit a group with a command setting the font (and another command setting the text direction for `\YHWH`) followed by some Unicode codepoints and another version to include a PDF graphic. It worked. But especially the two definitions for each of the three user macros suffered from noticeable code duplication.

Version 0.2 also added support for conversion to HTML with LaTeXML. But it did so through a separate binding and hence did not require changes to emo's implementation.

In contrast, for **version 0.3**, I integrated support for TeX4ht into the implementation and treated rendering Unicode codepoints *without* preceding font selection as a *third* backend. That increased variability and also the corresponding complexity of definitions. Notably, the internal `\emo@content` macro for rendering either Unicode codepoints, Unicode codepoints with font selection, or PDF graphic was defined thrice.

At the same time, the implementation also eliminated some variability by including entries for `\lingchi` and `\YHWH` in the emoji table. That made it possible for both macros to delegate to `\emo`—except when rendering Unicode codepoints *with font selection*. In the latter case, each macro renders glyphs in a different “language” and possibly writing system and hence also needs to select a different font.

For **version 0.4**, I introduced the debug package option, which instructs emo to make the bounding boxes for its output visible. To implement the option, I added code that wraps the result of a macro in an `fbox` three times. Clearly, that's two times too many.

That same version also added unit tests and, as a result of writing and debugging the unit tests, my TeX-fu improved considerably. So after releasing that version, I started looking for a more ergonomic approach to emo's implementation, discovered [LaTeX hooks](#), and switched to them for **version 1.0**.

### 5.5.2 A Skeleton of Render Hooks

`emo/render/before` To determine `emo`'s hooks, we consider the functional requirements of `emo`'s three `emo/render/emoji` macros and pay close attention to how those requirements converge and diverge:

`emo/render/chinese`

`emo/render/hebrew`

`emo/render/content`

`emo/render/after`

1. All three macros visually render one or more glyphs. Let's introduce the `emo/render/content` hook for that.
2. Each macro generates glyphs in a different "language" and writing system and thus may need to set the font, text direction, and so on. Let's introduce the `emo/render/emoji`, `emo/render/chinese`, and `emo/render/hebrew` hooks for that.
3. Just as in aspect-oriented programming, we sometimes need to track macro invocations (e.g., for the `index` package option) or modify macro results (e.g., for the `debug` package option). Let's introduce the `emo/render/before` and `emo/render/after` hooks for that.

Before version 1.0, `\lingchi` and `\YHWH` relied on a trailing `\xspace` to obviate the need for trailing backslashes before spaces. Continued support would require introducing another hook. If that hook is hardcoded into the user macro template, it adds unnecessary overhead for every invocation of `\emo`. If it is optional, then macro template instantiation becomes more complicated. Since both options increase complexity with little benefit, I dropped that feature with version 1.0.

Here are the `emo`'s render hooks:

```
60 \NewHook{emo/render/before}
61 \NewHook{emo/render/emoji}
62 \ifEmojiExtra
63 \NewHook{emo/render/chinese}
64 \NewHook{emo/render/hebrew}
65 \fi
66 \NewHook{emo/render/content}
67 \NewReversedHook{emo/render/after}
```

`\emo@makecommand` With the hooks declared, we are ready to define the internal `\emo@makecommand` macro, `\emo@command` which in turn defines each of the three user macros. Since those three macros differ

`\emo@key` in control sequence and font selection hook, `\emo@makecommand` takes two arguments.

`\emo@value` The user macro, in turn, takes one argument, the name of the entity to render. Since hooks have no arguments, we need a way for determining the current command, the key argument, and the resulting, rendered value. The implementation uses `\emo@command`, `\emo@key`, and `\emo@value` for just that purpose.

The rest is straight-forward: After defining `\emo@command` and `\emo@key`, the implementation tests whether that key is valid. If so, it invokes the `emo/render/before` hook, renders the content by invoking the font selection and content hooks inside a group, binds the result to `\emo@value`, invokes the `emo/render/after` hook, and yields the value. The group ensures that any sticky changes to font etc are contained within the macro invocation.

Admittedly, it took some trial and error to arrive at this version of `\emo@makecommand`. I started out with `\emo@makecommand` only assembling the body of each user macro. That required using `\edef` for the user macro and prefixing almost every token in

\emo@makecommand with \noexpand. I converged on the current, simpler implementation once I realized that I could pass the control sequence to be defined as an argument.

```

68 \def\emo@makecommand#1#2{
69   \newcommand*{#1}[1]{%
70     \def\emo@command{#1}%
71     \def\emo@key{##1}%
72     \emo@ifdef{##1}{%
73       \UseHook{emo/render/before}%
74       \def\emo@value{%
75         \begingroup%
76         #2%
77         \UseHook{emo/render/content}%
78         \endgroup%
79         \UseHook{emo/render/after}%
80         \emo@value}}}
```

### 5.5.3 The User Macros

\emo Render the named emoji either as Unicode or as a PDF graphic. To do so, instantiate the skeleton with the font selection hook for emoji.

```
81 \emo@makecommand\emo{\UseHook{emo/render/emoji}}
```

\lingchi Since \lingchi and \YHWH do not take arguments, they require a helper macro that provides the necessary argument to the macro generated by \emo@makecommand. Originally, both macros included a trailing \xspace in their expansion. While it would be possible to implement that with hooks based on \emo@command, it seems like an unnecessary complication, so I dropped that feature in version 1.0.

```

82 \ifEmojiExtra
83 \emo@makecommand\emo@lingchi{\UseHook{emo/render/chinese}}
84 \emo@makecommand\emo@YHWH{\UseHook{emo/render/hebrew}}
85 \newcommand*{\lingchi}{\emo@lingchi{lingchi}}
86 \newcommand*{\YHWH}{\emo@YHWH{YHWH}}
87 \fi
```

### 5.5.4 Activating the Hooks

So far, the three user-facing macros do nothing besides creating empty groups. We change that by activating the hooks as required by emoji's package options and the current engine. Conveniently, we only need to add to a hook if we need it to do something. In the following, we activate hooks in invocation order, from emoji/render/before to font selection hooks to emoji/render/content to emoji/render/after.

If the index option is enabled, create the necessary index and emit entries through the emoji/render/before hook. The hook uses the \emo@emit@index helper macro because stepping through [emo] requires three more \expandafter invocations.

```

88 \ifemo@index
89 \newindex{emo}{edx}{end}{Emoji Index}
90 \def\emo@emit@index#1{\index[emo]{#1}}
```

```

91 \AddToHook{emo/render/before}{%
92     \expandafter\emo@emit@index\expandafter{\emo@key}}
93 \fi

```

Next, if the backend uses fonts, set up font selection in the corresponding hooks. We only activate the `emo/render/chinese` and `emo/render/hebrew` hooks if the `extra` package option is enabled. Note that since Hebrew is written right-to-left, the font selection hook also set the text direction. This is safe to do because font selection and content are always enclosed in a group.

```

94 \ifemo@use@font
95 \newfontface\emo@font@emoji[Renderer=Harfbuzz]{NotoColorEmoji.ttf}
96 \AddToHook{emo/render/emoji}{\emo@font@emoji}
97 \ifEmojiExtra
98 \newfontface\emo@font@chinese{emo-lingchi.ttf}
99 \newfontface\emo@font@hebrew{LinLibertine_R.otf}
100 \AddToHook{emo/render/chinese}{\emo@font@chinese}
101 \AddToHook{emo/render/hebrew}{\emo@font@hebrew{textdir TRT}}
102 \fi
103 \fi

```

Next, we activate the `emo/render/content` hook to either emit Unicode codepoints or PDF graphics.

```

104 \ifemo@use@unicode
105 \AddToHook{emo/render/content}{\csname emoji@\emo@key\endcsname}
106 \fi
107 \ifemo@use@pdf
108 \AddToHook{emo/render/content}{%
109     \raisebox{-0.2ex}{%
110         \includegraphics[height=1em]{emo-graphics/emo-\emo@key}}}
111 \fi

```

Finally, if the `debug` package option is enabled, we wrap the rendered content in a frame box.

```

112 \ifemo@debug
113 \AddToHook{emo/render/after}{%
114     \let\emo@realvalue\emo@value%
115     \def\emo@value{\fbox{\emo@realvalue}}}
116 \fi

```

Et voilà. That's it!

```
117 </package>
```

## 6 LaTeXML Binding

To support conversion from LaTeX to HTML, `emo` includes a so-called binding for [LaTeXML](#). It effectively is a (much simplified) re-implementation of `emo`'s core functionality, only written in Perl against LaTeXML's API. The binding ignores the `index` option

and does not perform error checking on emoji names. If either is important to you, please compile the document with LaTeX first. Furthermore, the binding emits necessary Unicode codepoints only, without font annotations. If you want to specify fonts, please use a CSS fontstack.

Asking package authors to reimplement their packages for LaTeXML seems unreasonable to me. It leads to code duplication and places the maintenance burden on package authors. Yet, right after announcing `emo`, the question of LaTeXML support came up. LaTeXML includes the `latextml.sty` package, which defines `\iflatextml`. I would have used that command to make the three-line change to `emo.sty` necessary to support LaTeXML, except `latextml.sty` contains lots of other stuff that isn't needed. Always loading lots of macros only to detect LaTeXML slows down compilation and wastes memory. Since reimplementing `\iflatextml` would require a binding anyways, I just wrote a minimal binding. As I said, LaTeXML's approach is broken.

With that out of the way, let's get started:

```
1<(*latextml-binding>
```

The binding starts with an explicit preamble because `docstrip` does not allow for a redefinition of the starting characters of a line comment. It is followed by the Perl dependencies.

```
2## emo's LaTeXML binding.
3## (C) 2023 by Robert Grimm.
4## Released under LPPL v1.3c or later.
5use strict;
6use warnings;
7use LaTeXML::Package;
```

`\ifEmojiExtra` Next, we use raw TeX to declare the LaTeX package and the `\ifEmojiExtra` conditional. The binding does not require any other conditionals since it only runs under LaTeXML and does not support options other than `extra`.

```
8RawTeX(<<'EOTeX');
9\ProvidesPackage{emo}
10 [2023/99/99 v1.0 emo.ji for all (LaTeX engines)]
11 \newif\ifEmojiExtra
12 EOTeX
```

Option processing is almost trivial:

```
13 DeclareOption('extra', '\EmojiExtratrue');
14 DeclareOption('index', '');
15 DeclareOption('debug', '');
16 ProcessOptions();
```

`\EmojiBeginGroup` Define the five macros for parsing the emoji table. They are the same as those defined `\EmojiBeginSubgroup` by the package itself.

```
\DefineEmoji
\EmojiEndSubgroup17DefMacro('\EmojiBeginGroup{}', '');
\EmojiEndGroup18DefMacro('\EmojiBeginSubgroup{}{}', '');
```

```

19 DefMacro('`DefineEmoji{}{}',
20   '\expandafter\def\csname emoji@#1\endcsname{#2}');
21 DefMacro('`EmojiEndSubgroup{}{}', '');
22 DefMacro('`EmojiEndGroup{}', '');

Thusly prepared, read in the emoji table.

23 InputDefinitions('emo', type => 'def', noltxml => 1);

\emo Define the simplest possible version of \emo. It has no hooks, no error checking, no font
selection, and no support for index or debug. It does, however, expand the emoji table
entry.

24 DefMacro('`emo{}', '\csname emoji@#1\endcsname');

\lingchi If the \EmojiExtra conditional is enabled, provide similarly minimal re-definitions of the
\YHWH \lingchi and \YHWH macros.

25 if (IfCondition(T_CS('\ifEmojiExtra'))) {
26   DefMacro('`lingchi', "\x{51cc}\x{9072}");
27   DefMacro('`YHWH', "\x{05D9}\x{05D4}\x{05D5}\x{05D4}");
28 }

```

That's it for the binding, too.

29 </ltxml-binding>

## 7 Testing and Documenting Emo

The following package and documents help test emo across all major LaTeX engines and generate informative reports.

Alas, we first start an always false conditional to prevent execution of this code while generating the documentation.

```

30 <*scaffold>
31 \iffalse
32 </scaffold>

```

### 7.1 Emo's Support Package

1 <\*support>

The `emo-support` package defines high-level macros that make testing and documenting the `emo` package easier. At the same time, no attempt has been made to make `emo-support` reusable. The package exists only to support `emo`.

```

2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{emo-support}[2023/99/99 v1.0 Test & document emo]

```

There are no options to process.

```
4 \ProcessOptions\relax
```

We require `iftex` for determining the LaTeX engine, `multicol` for compact inventory listings, `xcolor` for appearances, and `emo` itself for its `\ifemo@use@unicode`, `\ifemo@use@font`, and `\ifemo@use@pdf` conditionals. This package also reuses `emo`'s font variables during test execution.

```
5 \RequirePackage{iftex}
6 \RequirePackage{multicol}
7 \RequirePackage{xcolor}
8 \RequirePackage[extra]{emo}
```

Since font selection across engines is a mouthful, we select the document fonts here inside the support package. That may include requiring `fontspec`, too.

```
9 \iftutex
10 \RequirePackage{fontspec}
11 \RequirePackage{libertinus}
12 \setmonofont{inconsolata}
13 \else
14 \RequirePackage{libertinus}
15 \RequirePackage{inconsolata}
16 \fi
```

### 7.1.1 Report Generation

`\enginename` I couldn't find an existing macro that provides the desired functionality, so we gotta round up the usual suspects to define the engine name.

```
17 \ifxetex
18 \def\enginename{XeTeX}
19 \else
20 \ifluatex
21 \def\enginename{LuaTeX}
22 \else
23 \ifpdftex
24 \def\enginename{pdfTeX}
25 \else
26 \def\enginename{\emph{unknown engine}}
27 \fi
28 \fi
29 \fi
```

`emo@canary@frameinner` Define frame and background colors for boundary boxes of sample text.

`emo@canary@frameouter`

`emo@canary@background`

```
30 \definecolor{emo@canary@frameinner}{HTML}{636366}
31 \definecolor{emo@canary@frameouter}{HTML}{48484A}
32 \definecolor{emo@canary@background}{HTML}{E5E5EA}
```

Adjust settings for `\fcolorbox` so that it serves as bounding box.

```
33 \setlength{\fboxrule}{0.5pt}
34 \setlength{\fboxsep}{0pt}
```

```
\emo@nobox The sample text may or may not show bounding boxes for words and emoji. It always  
\emo@wordbox shows the bounding box for the entire line.
```

```
\emo@linebox  
35 \newcommand\emo@nobox[1]{#1}  
36 \newcommand\emo@wordbox[1]{%  
37   \fcolorbox{emo@canary@frameinner}{white}{#1}}  
38 \newcommand\emo@linebox[1]{%  
39   \fcolorbox{emo@canary@frameouter}{emo@canary@background}{#1}}
```

```
\@samplertext Show a single line of text that makes use of emo's three macros. To help identify incor-  
\samplertext rect font metrics, spurious whitespace, and other issues, show the line's bounding box  
and, for the starred version, the bounding boxes for words and emoji, too.
```

```
40 \def\@samplertext#1{  
41   \emo@linebox{  
42     #1{It's} #1{\lingchi}:  
43     #1{Please}, #1{\YHWH}, #1{have} #1{mercy}  
44     #1{\emo{pleading-face}}! }%  
45   \vspace{1ex}}  
46 \newcommand*\samplertext{  
47   @ifstar{\@samplertext{\emo@wordbox}}{\@samplertext{\emo@nobox}}}
```

### 7.1.2 Emoji Inventory

The inventory makes use of custom section headings `\group` and `\subgroup`. Let's define their counters.

```
48 \newcounter{group}  
49 \newcounter{subgroup}[group]
```

```
\emo@subgrouprule Subgroup headings are particularly fancy. They show the subgroup name in small-caps  
\emo@subgroupstyle between horizontal rules. It's easiest to use separate paragraphs for each rule as well  
as the subgroup name. To prevent very ugly column breaks between them, we wrap  
everything in a minipage environment.
```

```
50 \def\emo@subgrouprule{\rule{\linewidth}{1pt}\par}  
51 \def\emo@subgroupstyle#1{  
52   \begin{minipage}{\linewidth}  
53     \emo@subgrouprule\par  
54     \rmfamily\textsc{#1}\vspace{-1.0ex}\par  
55     \emo@subgrouprule  
56   \end{minipage}}
```

```
\group We are ready to define the new sectioning commands. By using \@startsection, we  
\subgroup LaTeX's support for managing the space above and below the heading for free. That  
makes a huge difference, if the heading appears at the top of a page or columns.
```

```
57 \newcommand{\group}{%  
58   \@startsection{group}{1}{0pt}{  
59     {-3ex plus -1ex minus -0.2ex}{  
60     {3ex plus 1ex minus 0.2ex}}}
```

```

61      {\centering\rmfamily\Large\bfseries\itshape}
62 \newcommand{\subgroup}{%
63   \@startsection{subgroup}{2}{0pt}%
64     {-3.5ex plus -1.0ex minus -0.2ex}%
65     {2ex plus 0.5ex minus 0.2ex}%
66     {\emo@subgroupstyle}}

```

\emo@dotfilla The inventory formats emoji names flush left and the corresponding emoji flush right \emo@lineup with dots in between. Whereas \emo@dotfilla formats the dots in the middle, \emo@lineup@emoji \emo@lineup takes care of putting it all together into one paragraph. It also is careful to \emo@lineup@extra enable hyphenation for typewriter text. \emo@lineup@emoji and \emo@lineup@extra \emo@group@extra account for the differences between actual emoji and the two extra macros. Finally, \emo@group@extra names the extra group.

```

67 \def\emo@dotfilla{%
68   \unskip\nobreak%
69   \leaders\hbox{\hskip 0.1ex.\hskip 0.1ex}%
70   \hskip 1em plus 1fill\relax}
71 \def\emo@lineup#1#2{%
72   \noindent\raggedright%
73   \texttt{\hyphenchar\font=\`{-#1}%
74   \emo@dotfilla#2\par}
75 \def\emo@lineup@emoji#1#2{\emo@lineup{#1}{\emo{#1}}}
76 \def\emo@lineup@extra#1#2{\emo@lineup{#1}{\csname#1\endcsname}}
77 \def\emo@group@extra{extra}

```

\listinventory With that, we are well-prepared for defining the \listinventory macro. It does all formatting by reading the emoji table from `emo.def` with its own version of the file format macros. It even changes \DefineEmoji while reading `emo.def` to account for the final *extra* group. The resulting format lists emojis with their names under \group and \subgroup headings in three columns per page, all contained within a group.

```

78 \newcommand{\listinventory}{%
79   \begingroup
80   \setlength{\parskip}{0.3ex plus 0.2ex minus 0.1ex}
81   \let\DefineEmoji=\emo@lineup@emoji
82   \def\EmojiBeginGroup##1{%
83     \def\emo@group{##1}%
84     \ifx\emo@group\emo@group@extra%
85     \let\DefineEmoji=\emo@lineup@extra%
86     \fi%
87     \group*{##1}%
88     \def\EmojiBeginSubgroup##1##2{\subgroup*{##2}}
89     \begin{multicols}{3}%
90     \input{emo.def}%
91     \end{multicols}%
92   \endgroup}

```

### 7.1.3 Testing

```
\emo@canary@actual  Validating emo's macros turned out to be a bit trickier than I had expected. The obvious  
\emo@canary@expected  approach, fully expanding the macros and then comparing the results, doesn't work.  
While TeX does support eager expansion via, for example, \expandafter and \edef, it  
does so only for some macros.
```

Instead, we need to take a sneakier approach: Generate a box with the macro invocation and another box with the expected result and then compare the widths of the two boxes. While that is an incomplete comparison and hence cannot detect all bugs, it *can* detect any bug where the macro's output is shorter or longer than expected. That conveniently includes whitespace, which is one of the likeliest regressions.

We get started on that testing strategy by defining two box registers.

```
93 \newsavebox{\emo@canary@actual}  
94 \newsavebox{\emo@canary@expected}
```

\checkwidth The \checkwidth{\langle name\rangle}{\langle code\rangle}{\langle font\rangle}{\langle codepoints\rangle}{\langle file\rangle} macro can test emo's three user-facing macros across all engines. It captures the variability for doing so across all supported engines with its five arguments.

1. {\langle name\rangle} of the user macro being tested—e.g., `emo`;
2. {\langle code\rangle} to invoke the macro—e.g., `\emo{robot}`;
3. name of the command selecting {\langle font\rangle}—e.g., `emoji`;
4. expected Unicode {\langle codepoints\rangle}—e.g., `\char"1F916`;
5. {\langle file\rangle} name of the PDF graphic—e.g., `robot`.

\checkwidth's implementation starts by filling a box each with the actual macro content and the expected macro content. Since the actual content implicitly differs depending on backend, the latter must explicitly vary the expected content depending on backend.

```
95 \newcommand\checkwidth[5]{%  
96   \sbox{\emo@canary@actual{\#2}}%  
97   \ifemo@use@font%  
98     \sbox{\emo@canary@expected{  
99       \begingroup\csname\emo@font@\endcsname\#4\endgroup}%  
100    \else%  
101    \ifemo@use@unicode%  
102      \sbox{\emo@canary@expected{\begingroup\#4\endgroup}}%  
103    \else%  
104      \sbox{\emo@canary@expected{  
105        \raisebox{-0.2ex}{%  
106          \includegraphics[height=1em]{emo-graphics/emo-\#5}}}}%  
107    \fi%  
108  \fi%
```

It then generates the test report inside a box. The report states the macro name followed by a green check mark or red cross mark, depending on whether the two boxes have the

same width. If they do not, the red cross mark is followed by each box's content and width.

```

109  \mbox{%
110  \texttt{\char`\#1}\space%
111  \ifdim\wd\emo@canary@actual=\wd\emo@canary@expected%
112  \emo{check-mark-button}%
113  \else%
114  \emo{cross-mark}%
115  \fbox{\usebox{\emo@canary@actual}}\space%
116  \the\wd\emo@canary@actual\space%
117  \fbox{\usebox{\emo@canary@expected}}\space%
118  \the\wd\emo@canary@expected%
119  \fi}%

```

That's it for `emo`'s support package.

```
120 </support>
```

## 7.2 Engine Test and Report

```
1 <canary>
```

Each test report identifies the LaTeX engine and then shows the results of the width tests for `emo`'s three user-visible macros. If a test passes, the output only contains the macro name and a check mark. If a test fails, the output contains the macro name, the divergent box widths, and a cross mark. In the latter case, the width test results spill into the next line (at the least).

```

2 \documentclass[border=10pt, varwidth=6in]{standalone}
3 \usepackage{emo-support}
4 \setlength\parindent{0pt}
5 \setlength{\parskip}{1ex}
6 \begin{document}
7 \Huge
8 \emo{keycap-hash} \enginename: {\Large
9 \checkwidth{\emo{\robot}}{\emoji{\char"1F916}{robot}},
10 \checkwidth{\lingchi}{\lingchi{\chinese{\char"51CC}{\char"9072}{lingchi}}},
11 \checkwidth{\YHWH}{\YHWH{\hebrew}{%
12 \csname textdir\endcsname TRT\char"5D9\char"5D4\char"5D5\char"5D4%
13 }\{YHWH\}}}
14 \vspace{1ex}\par

```

Next is the sample text, first with and then without boundary boxes for words and emoji.

```

15 \sampletext*\par\sampletext
16 \end{document}

```

That's it for the tests and report.

```
17 </canary>
```

### 7.3 Simple Test Document

1<\*oneliner>

Not much to see here besides one line of content.s

```
2 \documentclass[border=10pt, varwidth=6in]{standalone}
3 \usepackage{emo-support}
4 \begin{document}
5 \Huge\sampletext
6 \end{document}
```

It's a wrap 😊

7</oneliner>

## Change History

0.1	General: Make initial release . . . . .	1	engines with <code>canary.tex</code> . . . . .	1
0.2	General: Add LaTeXML binding for conversion to HTML . . . . .	28	Introduce a simple unit testing framework . . . . .	30
	Prefix font and graphic files with “emo-” . . . . .	1	<code>\ifemo@debug</code> : Add debug option for drawing boundary boxes . . . . .	22
0.3	Support <code>pdftex</code> for extracting <code>emo.dtx</code> . . . . .	1	1.0	
	General: Include <code>\lingchi</code> and <code>\YHWH</code> in emoji table . . . . .	23	General: Drop <code>\textdir</code> from emoji table entry for <code>\YHWH</code> . . . . .	23
	Support <code>TeX4ht</code> for conversion to HTML . . . . .	1	Introduce high-level emoji table format that preserves Unicode grouping and order . . . . .	23
	<code>\lingchi</code> : Build on <code>\emo</code> by default . . . . .	27	Use LaTeX hooks for handling engines and options . . . . .	1
	<code>\YHWH</code> : Build on <code>\emo</code> by default . . . . .	27	<code>\checkwidth</code> : Since <code>xspace</code> is not used anymore, drop period from measured content . . . . .	34
0.4	General: Automate testing across		<code>\lingchi</code> : Drop use of trailing <code>xspace</code> . . . . .	27
			<code>\YHWH</code> : Drop use of trailing <code>xspace</code> . . . . .	27

## Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols	D
<code>\-</code> . . . . .	<code>\DeclareOption</code> . . . . .
<code>\@ifstar</code> . . . . .	<code>\def</code> . . . . .
<code>\@sampletext</code> . . . . .	<code>\def</code> . . . . .
<code>\@startsection</code> . . . . .	<code>\def</code> . . . . .
<code>\`</code> . . . . .	<code>\definecolor</code> . . . . .
<code>\{</code> . . . . .	<code>\DefineEmoji</code> . . . . .
<code>\}</code> . . . . .	<code>\documentclass</code> . . . . .
 <b>A</b>	
<code>\AddToHook</code> . . . . .	<code>\else</code> . . . . .
 <b>B</b>	
<code>\begin</code> . . . . .	<code>\emo</code> . . . . .
<code>\begingroup</code> . . . . .	<code>\emo/render/after</code> . . . . .
<code>\bfseries</code> . . . . .	<code>\emo/render/before</code> . . . . .
 <b>C</b>	
<code>\centering</code> . . . . .	<code>\emo/render/chinese</code> . . . . .
<code>\char</code> . . . . .	<code>\emo/render/content</code> . . . . .
<code>\checkwidth</code> . . . . .	<code>\emo/render/emoji</code> . . . . .
<code>\colorbox</code> . . . . .	<code>\emo/render/hebrew</code> . . . . .
<code>\csname</code> . . . . .	<code>\emo@canary@actual</code> . . . . .
	<code>\emo@canary@background</code> . . . . .
	<code>\emo@canary@expected</code> . . . . .
	<code>\emo@canary@frameinner</code> . . . . .
 <b>E</b>	

\emo@canary@frameouter . . . . .	30
\emo@command . . . . .	68
\emo@debugtrue . . . . .	7
\emo@dotfilla . . . . .	67
\emo@emit@index . . . . .	90, 92
\emo@error . . . . .	49, 58
\emo@error@bg . . . . .	49
\emo@error@fg . . . . .	49
\emo@font@chinese . . . . .	98, 100
\emo@font@emoji . . . . .	95, 96
\emo@font@hebrew . . . . .	99, 101
\emo@group . . . . .	83, 84
\emo@group@extra . . . . .	67, 84
\emo@ifdef . . . . .	55, 72
\emo@indextrue . . . . .	6
\emo@key . . . . .	68, 92, 105, 110
\emo@linebox . . . . .	35, 41
\emo@lineup . . . . .	67
\emo@lineup@emoji . . . . .	67, 81
\emo@lineup@extra . . . . .	67, 85
\emo@lingchi . . . . .	83, 85
\emo@makecommand . . . . .	68, 81, 83, 84
\emo@nobox . . . . .	35, 47
\emo@realvalue . . . . .	114, 115
\emo@subgrouprule . . . . .	50
\emo@subgroupstyle . . . . .	50, 66
\emo@use@fonttrue . . . . .	27
\emo@use@pdftrue . . . . .	29
\emo@use@unicodetrue . . . . .	23, 26
\emo@value . . . . .	68, 114, 115
\emo@wordbox . . . . .	35, 47
\emo@YHWH . . . . .	84, 86
\EmojiBeginGroup . . . . .	17, 42, 82
\EmojiBeginSubgroup . . . . .	17, 43, 88
\EmojiBeginSubroup . . . . .	42
\EmojiEndGroup . . . . .	17, 42
\EmojiEndSubgroup . . . . .	17, 42
\EmojiExtratrue . . . . .	5, 13
\emph . . . . .	26
\end . . . . .	6, 16, 56, 91
\endcsname . . . . .	12, 20, 24, 45, 56, 76, 99, 105
\endgroup . . . . .	78, 92, 99, 102
\enginename . . . . .	8, 17
\expandafter . . . . .	20, 45, 92
<b>F</b>	
\fbox . . . . .	115, 115, 117
\fboxrule . . . . .	33
\fboxsep . . . . .	34
\fcolorbox . . . . .	37, 39
\fi . . . . .	16, 22, 27, 28, 29, 30, 31, 34, 38, 41, 59, 65, 86, 87, 93, 102, 103, 106, 107, 108, 111, 116, 119
\font . . . . .	73
<b>G</b>	
\group . . . . .	57, 87
<b>H</b>	
\hbox . . . . .	69
\HCode . . . . .	14
\hskip . . . . .	69, 70
\Huge . . . . .	5, 7
\hyphenchar . . . . .	73
<b>I</b>	
\ifcsname . . . . .	56
\ifdefined . . . . .	14
\ifdim . . . . .	111
\ifemo@debug . . . . .	2, 112
\ifemo@index . . . . .	2, 36, 88
\ifemo@use@font . . . . .	10, 32, 94, 97
\ifemo@use@pdf . . . . .	10, 39, 107
\ifemo@use@unicode . . . . .	10, 101, 104
\ifEmojiExtra . . . . .	2, 8, 25, 62, 82, 97
\iffalse . . . . .	31
\ifluatex . . . . .	15, 20, 25
\ifpdftex . . . . .	23
\iftutex . . . . .	9
\ifx . . . . .	84
\ifxetex . . . . .	17
\includegraphics . . . . .	106, 110
\index . . . . .	90
\input . . . . .	48, 90
\itshape . . . . .	61
<b>L</b>	
\Large . . . . .	8, 61
\leaders . . . . .	69
\let . . . . .	81, 85, 114
\linewidth . . . . .	50, 52
\lingchi . . . . .	10, 25, 42, 82
\listinventory . . . . .	78
<b>M</b>	
\mbox . . . . .	109
\MessageBreak . . . . .	17, 18, 19
<b>N</b>	
\NeedsTeXFormat . . . . .	2
\newcommand . . . . .	35,
	36, 38, 46, 57, 62, 69, 78, 85, 86, 95
\newcounter . . . . .	48, 49
\newfontface . . . . .	95, 98, 99
\NewHook . . . . .	60, 61, 63, 64, 66
\newif . . . . .	2, 3, 4, 10, 11, 11, 12
\newindex . . . . .	89
\NewReversedHook . . . . .	67
\newsavebox . . . . .	93, 94
\nobreak . . . . .	68
\noindent . . . . .	72

<b>P</b>	
\PackageError	16
\PackageWarning	57
\par	14, 15, 50, 53, 54, 74
\parindent	4
\parskip	5, 80
\ProcessOptions	4, 8
\ProvidesPackage	3, 9
<b>R</b>	
\raggedright	72
\raisebox	105, 109
\relax	4, 8, 70
\RequirePackage	5, 6, 7, 8, 9, 10, 11, 13, 14, 15, 33, 35, 37, 40
\rmfamily	54, 61
\rule	50
<b>S</b>	
\sampletext	5, 15, 40
\sbox	96, 98, 102, 104
\setlength	4, 5, 33, 34, 80
\setmonofont	12
\space	110, 115, 116, 117
\string	57
\subgroup	57, 88
<b>T</b>	
\textcolor	53
\textdir	101
\textsc	54
\textsf	54
\texttt	54, 73, 110
\the	116, 118
<b>U</b>	
\unskip	68
\usebox	115, 117
\UseHook	73, 77, 79, 81, 83, 84
\usepackage	3, 3
<b>V</b>	
\vspace	14, 45, 54
<b>W</b>	
\wd	111, 116, 118
<b>X</b>	
\x	26, 27
<b>Y</b>	
\YHWH	11, 25, 43, 82