



Dozenal Jottings

The new official newsletter of the Dozenal Society of America

Issue Two • September 13_z, 1207_z (September 15_d, 2026_d)

Editor: John Volan • Member 418_z(596_d) • editor@dozenal.org



DOZENAL IPHONE APP SUITE

TABLE OF CONTENTS

DOZENAL IPHONE APP SUITE	1
DozTIMER	1
DozSTOPWATCH	2
DozCLOCK	3
DozCLOCK WIDGET	4
DozANALOGCLOCK	5
DozALARM	6
DozALARM WIDGET	7
DozWEATHER	8
DozWEATHER WIDGET	9
DozWEATHERPLUS	10
CONCLUSION	14
DSA BOARD MEETING	14
IPHONE APPS DISCUSSION	14
PLANS FOR ANNUAL MEETING	14
NEXT BOARD MEETING	15
GOODBYE “STADEGREES”	15
ADJOURNMENT	15
APPEAL TO MEMBERS	15
WEBSITE UPDATES	16
NEW MEMBERS	17
AD: PRECISION DOZENAL TOOLS	18

The *Dozenal Society of America* is a voluntary, nonprofit educational corporation organized for the conduct of research and education of the public in the use of *dozenal* (also called *duodecimal* or *base-twelve*) in calculations, mathematics, weights and measures, and other branches of pure and applied science.

The DSA does not endorse any particular notation or pronunciation scheme for alternative bases. However, this publication defaults to using the Pitman 7 and 8 as digits ten and eleven, but may also use the “Computerese” transdecimal digits A through Z to represent larger bases. To distinguish numerals in different bases as compactly and equitably as possible, this publication uses single-character subscript base annotations, e.g. d = decimal, z = dozenal, x = hexadecimal, etc.

GREETINGS dozenal fans! The Dozenal Society of America is pleased to announce a suite of new dozenal apps for iPhone. They were developed by our long-time Canadian friend and board member Paul Rapoport, Member Number 230_z(324_d), with the help of an AI named Claude. Even with AI help, this represents several months of dedicated effort on Paul’s part, for which he deserves considerable appreciation.

The apps are all free to install from the App Store. Here is how they appear on a Home screen once installed:



This article will describe each of these apps in the order shown, roughly from simplest to most complicated. This will include how to install and operate them, and most of their features. Three of these come with widgets you can install right on the Home screen. We will demonstrate those as well. You should be able to use this article as a user’s manual.

All the apps include a link to **Dozenal Suite** guide documentation, which is located at <https://dozenal.ca/suite/>.

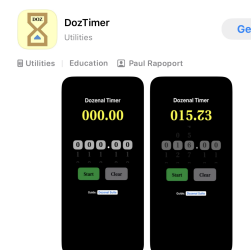
All the apps use the Pitman ten and eleven digits (7 and 8). Due to the limitations of mobile phone displays, no base annotations are provided. However, given the prominent “Dozenal” in the title of each app, the base can be inferred. (Paul’s documentation does provide annotations.)

For dozenal units of measure, these apps make use of the **Primel** metrology. (For more information, see the introductory article at <https://dozenal.org/article-volan-primel-metrology>.) In the case of time, Primel’s units are identical to the original Do-Metric units established by the DSA founders back in 1161_z

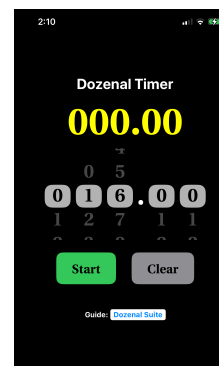
(1945_d), except that Primel gives them all different names.

DozTIMER

If you search for “DozTimer” at the App Store, you will now find this entry:



Install this to get a simple count-down timer:



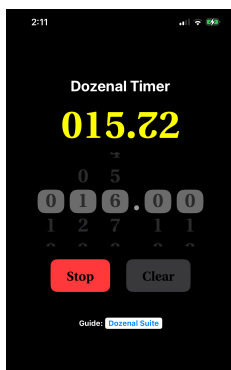
This lets you count down any amount of time up to one day, with five dozenal digits of precision:

- The top digit counts **uncia-days** (u↓dy) or “dwells” (what Do-Metric called “duors”) which are 2 hours each. (Formally, these are **prime-pentqua-timels**, abbreviated □p↑tmℓ.)
- The second digit counts **bicia-days** (b↓dy) or “breathers” (what Do-Metric called “temins”) which are ten minutes each. (Formally, these are **prime-quadqua-timels**, abbreviated □q↑tmℓ.)

- The third digit counts **tricia-days** (t↓dy) or “**trices**” (what Do-Metric called “**minettes**”), which are each 50_d seconds. (Formally, these are **prime-triqua-timels**, abbreviated □t↑tmℓ.)
- The fourth digit counts **quadcia-days** (q↓dy) or “**lulls**” (what Do-Metric called “**grovics**”), which are 4.16_d seconds each. (Formally, these are **prime-biqua-timels**, abbreviated □b↑tmℓ.)
- The lowest digit counts **pentcia-days** (p↓dy) or “**twinklings**,” (what Do-Metric called “**dovics**”), which are about a third of a second each ($0.347\overline{2}_d$ s). (Formally, these are **prime-unqua-timels**, abbreviated □u↑tmℓ.)

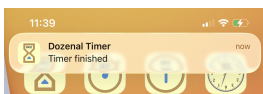
The placement of the radix point effectively scales the count to trices and fractional trices.

Use the rollers in the middle to set the amount of time to count down. Here, we’ve selected 16_z trices (15_d minutes). Click the green **Start** button, and the timer will begin counting down the twinklings. The **Start** button becomes a red **Stop** button:



Pressing this pauses the count-down and turns it back into a **Start** button. Pressing **Start** again resumes the count-down.

When the count-down reaches zero, a tone sounds. An alert message also pops up, which you can see even if you’ve swiped the app into the background:



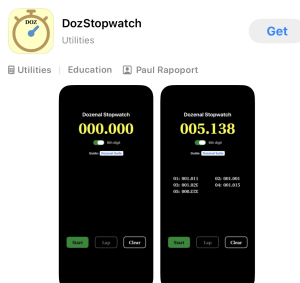
Clicking the pop-up will bring you back into the app. Press **Stop** to dismiss the tone.

The rollers in the middle remain locked to your original setting. Pressing

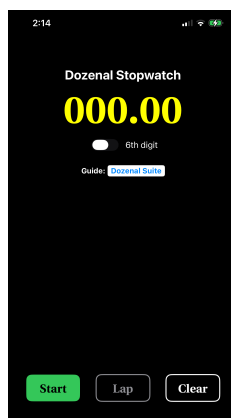
Start again repeats the count-down starting from the same setting. Press **Clear** to zero the rollers. This unlocks them so you can pick a different amount of time.

DozSTOPWATCH

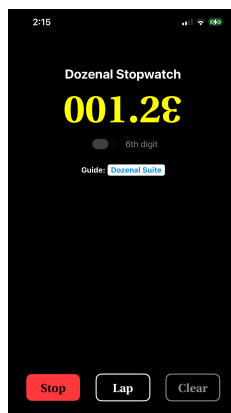
If you search for “DozStopwatch” at the App Store, you will find this entry:



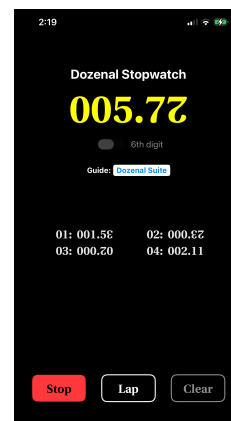
Install this to get a basic dozenal stopwatch function, scaled to trices and fractional trices, down to the twinkling:



Press **Start** to begin counting elapsed time. The **Start** button becomes a red **Stop** button:

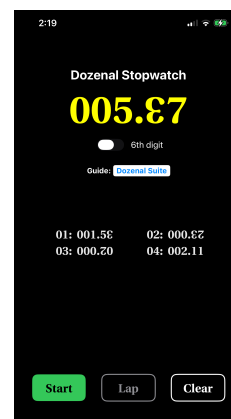


The **Lap** button records a snapshot of elapsed time in the space below:



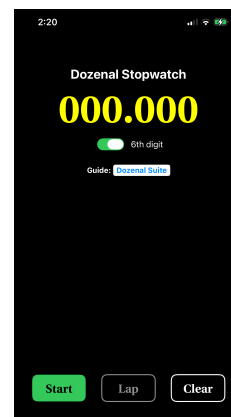
This may be repeated up to 60_z (72_d) times. Note that each lap time is the time since the previous lap, not the cumulative time.

Pressing **Stop** pauses counting elapsed time and turns it back into a **Start** button:



Pressing **Start** again resumes counting elapsed time. Pressing **Clear** resets the stopwatch to zero and erases all the recorded laps.

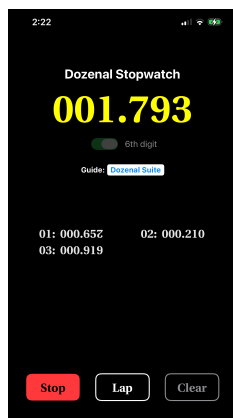
Note the toggle button labeled **6th digit**. Sliding this to the right extends the precision of the stopwatch to six digits:



Press **Start** and the stopwatch will count elapsed time down to the

hexcia-day or “vibe” (what Do-metric called a “vic”), which is equivalent to 28.93518_d milliseconds. This is actually Primel’s coherent unit of time for physics, the **prime-timel** (abbreviated $\square\text{tm}\ell$).

In this mode, lap times will also be recorded with “vibe” precision:

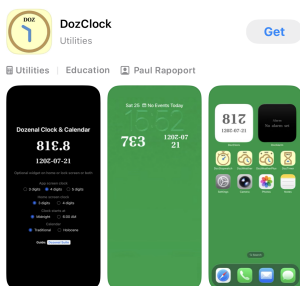


This “vibe” digit cycles far too fast for the human eye to track. However, this is not unreasonable. Professional stopwatches used at international sporting events such as the Olympics typically operate even faster, providing precision down to the hundredth of a second (10_d milliseconds), or even the millisecond.

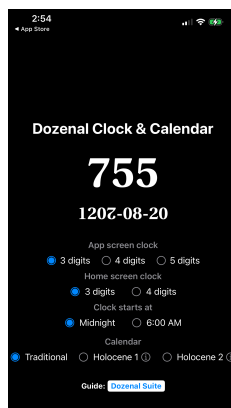
The “vibe” digit changes at a frequency of 34.56_d hertz (which is one **prime-frequenciel** ($1 \square\text{fq}\ell$)). As a sound, this would actually be on the threshold of audibility. In fact, the Do-Metric name “vic” stood for “vibration of C# in octave zero” because that is the standard note closest to that frequency. Primel prefers to use known English words rather than portmanteaus, so calls this simply a “vibe.”

DozClock

If you search for “DozClock” at the App Store, you will now find this entry:



Install this to get a digital clock and calendar function:



The prominent three-digit numeral in the center is the time of day in tricia-days or “trices,” expressed in dozenal. This is a “diurnal” clock, because it cycles only once a day, from 000_z to $\text{E}\text{E}\text{E}_z$ trices. For example, the time shown above, 755_z trices, corresponds to approximately $2:54_d$ P.M.

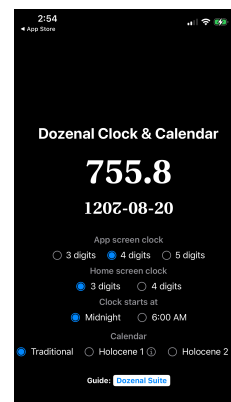
This is much simpler than traditional sexagesimal clocks, where you must contend with a complex structure: binary for the choice between A.M. and P.M., then base twelve for the hours in a half-day, then base sixty for the minutes in an hour, yet all a done using base *ten* to express the numerals.

Instead, you have a *single* numeral in a *single* base—dozenal—expressing a simple fraction of a day. Even if we add more digits to make the time more precise, it’s still just a single numeral.

In fact, this is exactly the concept behind “metric” time as attempted during the French Revolution, except that this is *dozenal* metric rather than *decimal* metric. This is how the DSA founders preferred to specify time using their Do-Metric system, back in the earliest days of the Society (albeit using different names for all the units).

Underneath this, in a smaller font, is the traditional Gregorian calendar date, in ISO 8601 big-endian format (YYYY-MM-DD), again expressed in dozenal. (This snapshot was taken on August 24_d , 2026_d , otherwise known as August 20_z , 1207_z .)

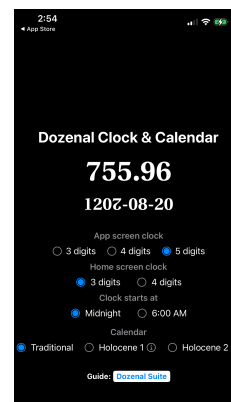
The lower half of the display contains controls for configuring the clock. The default setting for the **App screen clock** is **3 digits**. However, we can increase this to **4 digits**, to express the time with “lull” accuracy:



Like the other apps, DozClock scales time to the trice, so the lulls are treated as fractions of a trice. For example, time 755.8_z trices corresponds to approximately $2:54:43_d$ P.M. However, this does not increase the complexity of the time display. It doesn’t introduce a “lull field” the way traditional time adds an entire base-sixty field for seconds. It’s simply appending another digit onto a single number, with an arbitrary choice for where to place the radix point.

In principle, we could leave off the radix point and count time of day in whole lulls. (Here it would be 7558_z lulls.) Or we could even shift the radix point all the way to the left and simply look at this as a fractional quantity of days. (Here it would be 0.7558_z days.) However, DozClock only provides scaling to trices. That said, it’s a reasonable choice.

We can even go to **5 digits**, to express the time with “twinkling” accuracy:



Again, this doesn’t add any more complexity, just more precision. For example, the time 755.96_z trices corresponds to approximately $2:54:49_d$ P.M.

The default setting for the **Calendar** display is **Traditional**, which means the traditional Gregorian calendar, except expressed in dozenal rather than decimal. The default setting for when to

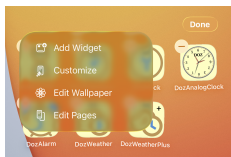
start the day is **Midnight**. However, to support his experimentation with clocks and calendars, Paul has included options to start the day at **6:00_d A.M.** (nominal average sunrise), as well as to use his “Holocene” calendar, whose epoch is set to approximately the end of the last Ice Age, and which uses an astronomically-based way of breaking up the year into months. He offers two version of this: **Holocene 1** starts the year at the December solstice, while **Holocene 2** starts the year at the March equinox. (For more information, see his documentation **About the Dozenal Holocene Calendar** at <https://clocks.dozenal.ca/pdf/dozenal-calendar.pdf>.)

DozClock Widget

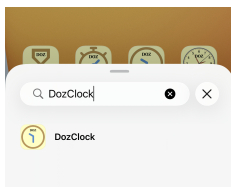
DozClock also provides a widget that can be displayed directly on the Home screen. To add it, press and hold on the background of any Home screen page until the icons acquire edit decorations and start jiggling:



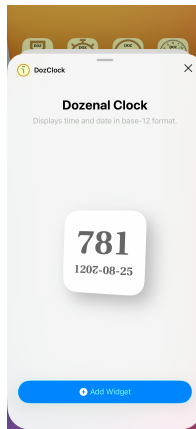
When the **Edit** button appears in the upper left hand corner, press it to get the Home screen edit menu:



Select **Add Widget** to bring up the **Search Widgets** dialog and type “DozClock”:



Select the **DozClock** icon in the list to bring up this dialog:

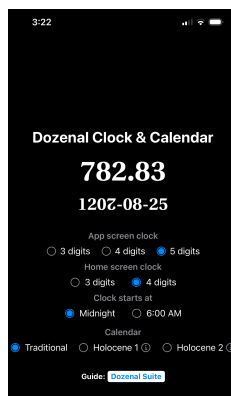


Press the **Add Widget** button at the bottom. The end result will be to place a DozClock widget on your Home screen:

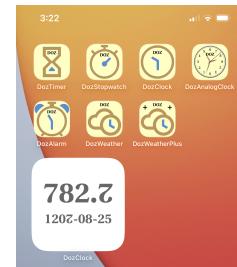


If you don’t like where it drops the widget, press and hold on the icon to grab it, then drag it to where you’d like. If other app icons got shuffled in the process, you can do the same to those.

The widget can be configured in the DozClock app. The default setting for **Home screen clock** is **3 digits**, giving trice precision. (For example, a time of 782_z trices corresponds to approximately 3:22 P.M.) To change this, press on the widget to bring up the DozClock app:



Under **Home screen clock**, you can select **4 digits**, with the following result on the widget on the Home screen:



This adds a lull digit (as a fraction of a trice). For example, the time of 782.7_z trices corresponds to approximately 3:22:22_d P.M. Unfortunately, because of Apple UI restrictions, there is not enough room to display 5 digits in the widget.

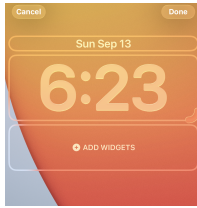
DozClock also provides two widgets for the iPhone Lock Screen: one widget for time-of-day and another for calendar date. First, press the right-side button on your iPhone to lock the screen:



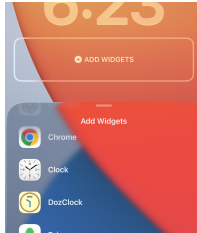
Press and hold on the background to enter the Lock Screen Gallery:



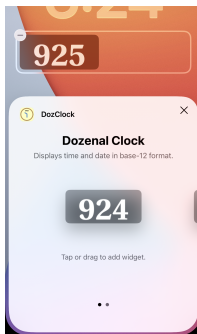
Swipe left and right to choose a Lock Screen, then press the **Customize** button to enter the Lock Screen Editor:



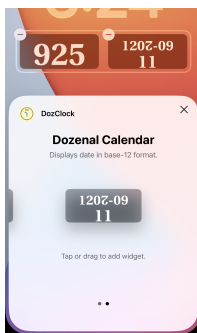
Press the **Add Widgets** button and scroll down the list until you find the Doz-Clock widget:



Press DozClock to open the Doz-Clock dialog. The first page of the dialog shows DozClock's Lock Screen widget for time. Tap it to add it to the Lock Screen:



Swipe left to the next page to show DozClock's Lock Screen widget for date. Tap it to add it to the Lock Screen:



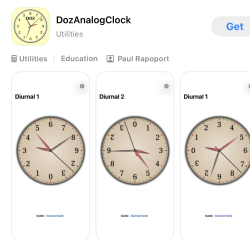
Tap the background to dismiss the dialog. Then press **Done** to return to the Gallery. Finally, swipe up to dismiss the Gallery and return to the Lock Screen:



The end result is that your Lock Screen now contains widgets displaying the current time in dozenal trices and the current date in ISO-8601 format in dozenal. Apple's limits on Lock Screen widgets are even more stringent than for Home Screen widgets. So the timestamp is limited to 3 digits at that font size.

DozANALOGClock

As an added bonus, Paul has ported his very nice online dozenal analog clock to iPhone. This turned out to be something Claude AI could do quite easily. If you search for "DozAnalogClock" at the App Store, you will now find this entry:



Install and run this to get the following analog clock:



Like a traditional clock, the dial is divided into twelve major markings, la-

beled with numerals. Unlike a traditional clock, the numerals are single digits in dozenal. Even the top numeral is just a zero rather than a twelve or dozen. Dozenal base is more natural for a clock; there is no need for two-digit numerals on any of the major divisions.

Unlike a traditional clock, the small markings between two numerals divide that sector into a dozen parts, rather than five. This makes for a gross fine markings around the dial.

By default, four hands are shown, rather than only two or three. Each spins a dozen times faster than the one before it, so each can encode a single digit of dozenal diurnal time at a glance. However, because they all spin smoothly, each can also encode the next digit, by looking closely at the fine dial markings.

- The short red hand is the **dwell** hand (not an hour hand). It rotates once per day (not half-day). It counts a dwell (two hours) at each numeral, and a breather (ten minutes) at each fine marking.
- The long red hand is the **breather** hand. It rotates once per dwell (not per hour). It counts a breather (ten minutes) at each numeral, and a trice (fifty seconds) at each fine marking.
- The thicker gray hand is the **trice** hand. It rotates once per breather (ten minutes). It counts a trice (fifty seconds) at each numeral, and a lull ($4.1\overline{6}_d$ seconds) at each fine marking.
- The thin gray hand is the **lull** hand. It rotates once per trice (fifty seconds). It counts a lull at each numeral, and a twinkling (about a third of a second) at each fine marking.

Like the digital DozClock app, this is a **Diurnal** clock, so there is no need for a context of A.M. or P.M. Since noon is at 6 dwells, A.M. (*ante meridiem*) is just the half of the dial from 0 to 6; P.M. (*post meridiem*) is the half from 6 to 0.

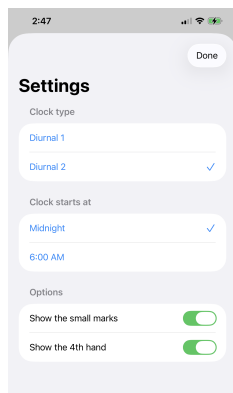
By default, the day starts at **Midnight**, and the clock is oriented with 0 at the top. So it's midnight when the dwell hand is pointing up at 0 dwells, noon when it's pointing down at 6 dwells, average sunrise ($6:00_d$ A.M.) when it's pointing to the right at 3 dwells, and average sunset ($6:00_d$ P.M.) when it's pointing to the left at 9 dwells.

So for instance, the above clock would be read as seven dwells, four breathers, nine trices, zero lulls, and

(squinting at the fine marks) ten twinklings. That would make the time of day $749.0\overline{7}_z$ trices since midnight, or approximately $2:47:33_d$ P.M.

However, notice that this clock configuration is labeled **Diurnal 2**. That's because, even though it follows traditional expectations for the orientation of a numbered dial (with 0 at the top), nevertheless it's not the orientation favored by many dozenalists that have thought about clocks in depth. Since the time of the DSA founders (and even before), dozenalists have felt it more natural to place 0 at the *bottom* of a diurnal dial.

DozAnalogClock supports this orientation as **Diurnal 1**. You can configure this by pressing the gear icon on the top right to open the **Settings** page:



Select **Diurnal 1** in the **Clock type** section, and then press the **Done** button in the upper right corner to return to the analog clock display:



Guide: Dozenal Suite

The rationale for this is that this way the dwell hand always points roughly in the direction where the sun would be, if one is looking toward the equator, and hence towards sun's ecliptic path. In the Northern Hemisphere, that means looking towards the south. At midnight (0 dwells) the sun is at nadir, beneath the

viewer's feet, eclipsed by the body of the Earth. At average sunrise (3 dwells), the sun is east, to the left. At noon (6 dwells), the sun is at zenith, directly above. At average sunset (9 dwells), the sun is in the west, to the right. Thus, the sun would be observed to rotate clockwise through the sky.

The above image shows the time as now $74\overline{7}.23_z$ trices since midnight, approximately equivalent to $2:48:29_d$ P.M. (a little more than a trice, or almost a minute, after the previous image).

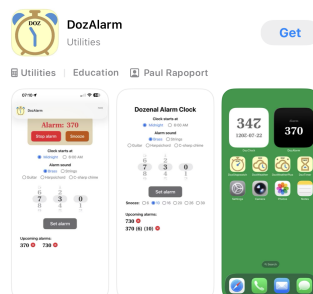
(In the southern hemisphere, one would look northward toward the equator, and see the mirror-image of the above: Midnight is still nadir and noon is still zenith, but average sunrise and east are to the right, and average sunset and west are to the left. The sun would be observed to rotate *counter*-clockwise through the sky. Presumably one would need a reverse-rotating clock; perhaps that would be called a "*counter*-clock!" However, DozClock does not provide that option. The best folks south of the equator can do is face *away* from the sun and imagine its path behind their backs!)

On the **Setting** page, the **Clock starts at** at setting defaults to **Midnight**, but may also be configured so that the day starts at **6:00_d A.M.** (average sunrise). This places average sunrise at 0 dwells, noon at 3 dwells, average sunset at 6 dwells, and midnight at 9 dwells. Note that this doesn't affect the orientation of midnight in either of the **Diurnal** configurations. **Diurnal 2** will still have midnight at the top (labeled 9), with 0 (sunrise) to the right, whereas **Diurnal 1** will still have midnight at the bottom (labeled 9), with 0 (sunrise) to the left.

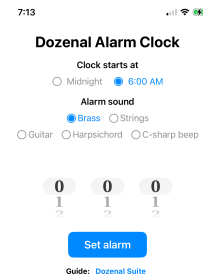
The **Settings** page also provides options to toggle whether to show the small marks on the dial, as well as the fourth hand.

DOZALARM

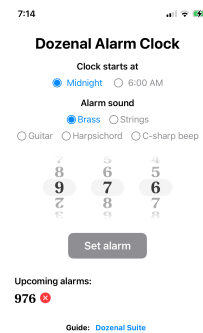
If you search for "DozAlarm" at the App Store, you will now find this entry:



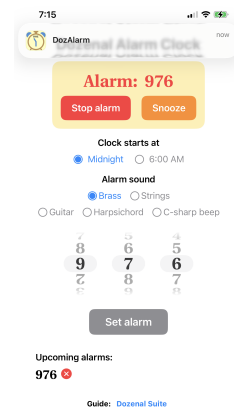
Install this to get a basic alarm function in dozenal:



Use the rollers to pick a time in trices to set an alarm for, then press **Set Alarm**. For instance:



This has set an alarm for 976_z trices, which is equivalent to $7:15_d$ P.M. In this snapshot, the current time is already 975_z trices and some change (just past $7:14_d$ P.M.), so the alarm time is only lulls away.



When the scheduled time arrives, DozAlarm generates an alarm tone (which may be an entire musical excerpt) based on the selection in the **Alarm sound** section in the app. The app adds a control box in yellow indicating which scheduled alarm is sounding, with buttons in red to **Stop alarm** or **Snooze**. It also pops up a transparent **DozAlarm** alert dialog, even if the app has been swiped into the background. You can press that to get into the app.

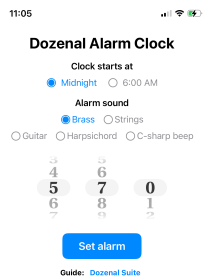
Pressing **Stop alarm** dismisses the alarm tone and removes the yellow control box. However, it does not remove the alarm time from the **Upcoming alarms** list, so the alarm will alert at the same time on subsequent days.

If you do nothing, after a bit more than a trice (about a minute) the alarm sound will stop playing on its own, although the alert box will continue to appear.

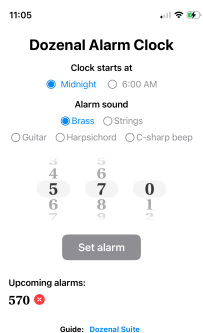
To delete an alarm, press the red X next to it.

DOZALARM WIDGET

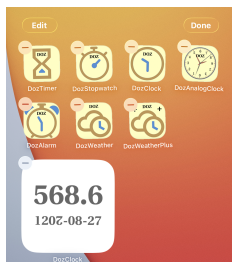
DozAlarm includes a widget that can be installed on the Home screen. It displays the next pending alarm (the alarm soonest to sound). To demonstrate this, first let's use the app to set a pending alarm.



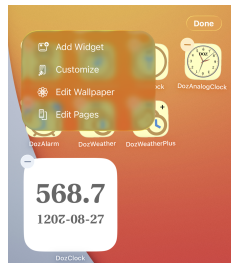
Here, the current time is 566_z trices (11:05_d A.M.). Let's add an alarm for 570_z trices (11:10_d A.M.):



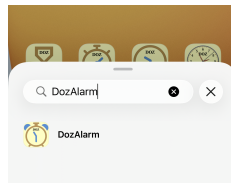
On a Home screen, press and hold the background until it switches to edit mode:



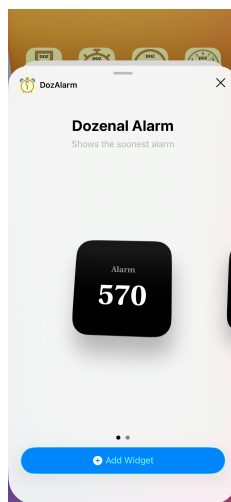
Press the **Edit** button to bring up the edit menu:



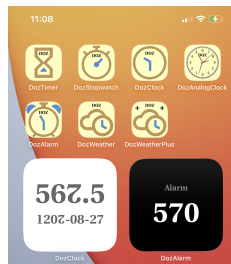
Select **Add Widget** to bring up the **Search Widgets** dialog, then type "DozAlarm":



Select the **DozAlarm** icon to bring up this **Add widget** dialog:

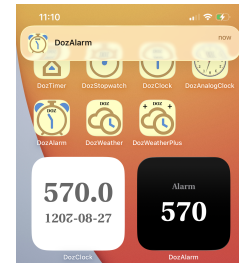


As you can see, the depiction of the widget is already displaying our alarm time of 570_z trices. Press **Add Widget** to add it to your Home screen. Here's the result (after shuffling all your icons back into your intended locations):

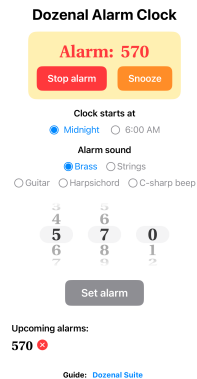


It's useful to place the **DozAlarm** widget next to the **DozClock** widget.

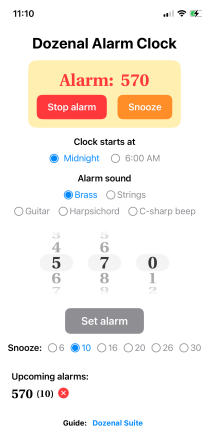
When the alarm goes off, the selected melody will sound, and a **DozAlarm** alert box will pop up:



Press the alert box or the widget to bring up the app:



Let's take this opportunity to demonstrate the snooze function. Press the **Snooze** button to yield:

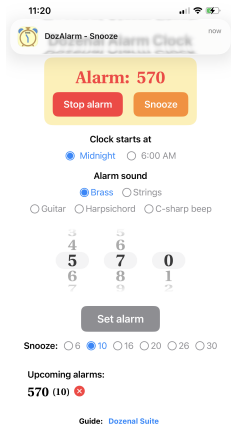


This stops the alarm sound, disables the **Snooze** button, and pauses the alarm for a default of a dozen trices (ten minutes), so you can literally "take a breather." However it also brings up **Snooze** options that allow you to change the snooze time to any increment of 6 trices (5 minutes) to up to 30_z trices (30_d minutes), or a quarter-dwell. Your chosen snooze time in trices will appear in parentheses next to your alarm time in the list.

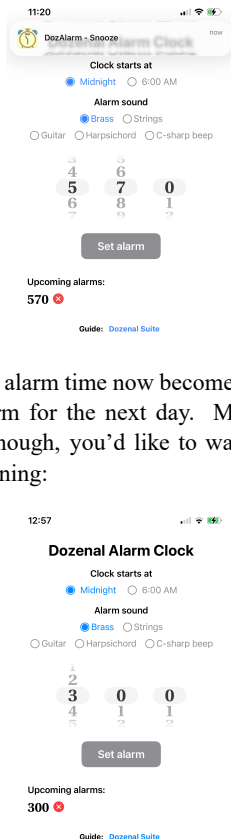
When the alarm sounds again, the alert box will read **DozAlarm - Snooze**:



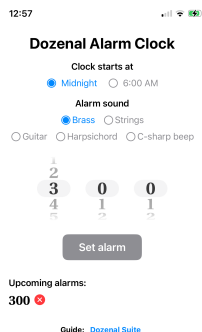
As before you can press the alert box or the widget to bring up the app:



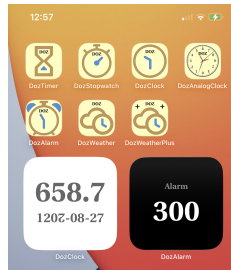
You can press **Snooze** again if you like, in which case a second snooze time in trices will appear next to the first. Here, let's press **Stop alarm** to dismiss it:



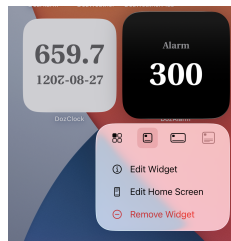
The alarm time now becomes a pending alarm for the next day. More than likely though, you'd like to wake up in the morning:



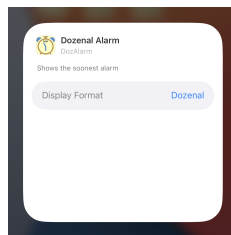
Here we've set an alarm for 3 dwells, which is a good time to get up in order to make it to work before the customary 460_z trices. Since this is now the next pending alarm, it's reflected on the wid-
get:



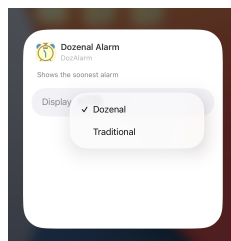
If you're still getting used to dozenal diurnal time, and want to reassure yourself that's the right time, press and hold the widget until this menu pops up:



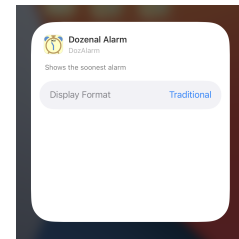
Select **Edit Widget** to bring up this dialog box:



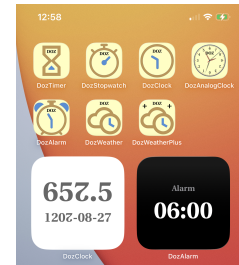
This shows that the **Display Format** of the widget is currently **Dozenal**. Press that to get a menu:



So the widget can show the next pending alarm in either **Dozenal** (trices) format, or **Traditional** format. Let's select **Traditional**:



Then press the background to dismiss the dialog:



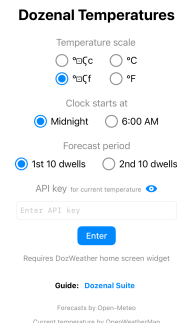
Now the widget displays the next pending alarm as 06:00_d. This is in 24_d-hour time, so you can be assured that this means 6 A.M. (6 P.M. would read as 18:00_d.) But you will likely want to edit the format back to **Dozenal**.

DozWeather

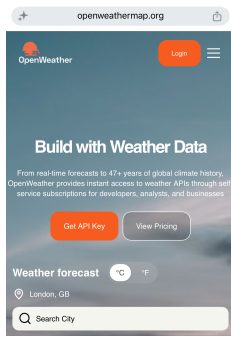
If you search for "DozWeather" at the App Store, you will find this entry:



Install this to get a basic **Dozenal Temperatures** app:

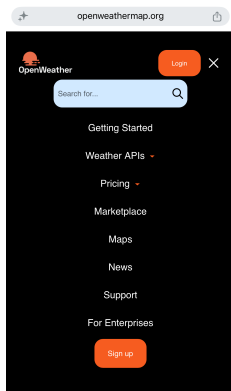


This app makes use of a service called **Open Weather Map** to provide current meteorological data. You will need to obtain a free API key from <https://openweathermap.org/>:

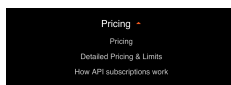


However we recommended that you acquire a **One-Call 4.0** subscription (You will need it when we get to the next app in the list.) It's practically free, because it gives you 1,000_d API calls per day for free, and anything above that is charged at a rate of \$0.15_d per 100_d calls. It is extremely unlikely for your usage of the Dozenal Suite apps to generate that many API calls. (These subscriptions are generally used by third-party services that do bulk API calls for large numbers of customers.)

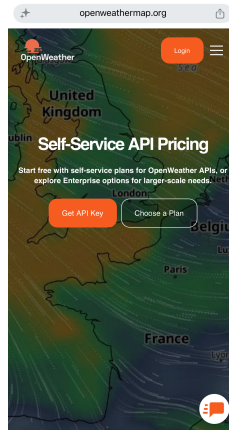
To acquire a subscription, press the hamburger icon in the top right corner to bring up the menu:



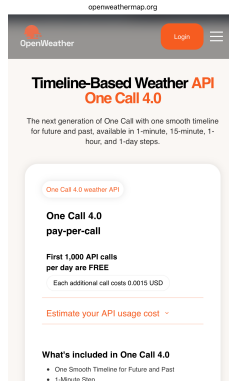
Expand the **Pricing** section:



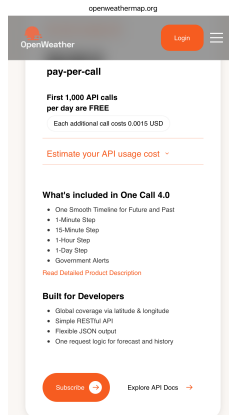
Press the **Pricing** menu item to bring up the **Self-Service API Pricing** page:



Scroll down on this page until you reach the section about **One Call 4.0**:

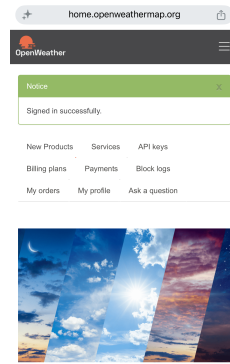


You can read the terms there, and if you are satisfied, scroll further to find the **Subscribe** button:

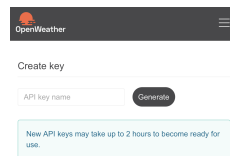


This will lead you to create an account and a password, and provide a credit card for payment. (However, it's very unlikely you will ever get charged.)

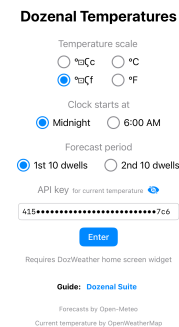
Once you have subscribed and created your account, sign in to reach your home page:



Press the **API keys** link to reach a page containing all your named keys. To create a key, scroll down to this section:



Once you have obtained an API key, copy it, then go back to the DozWeather app and paste it into the **API key for current temperature** box:



For security reasons, it's recommended that you press the eye icon (as you see here), to obscure your API key from prying eyes.

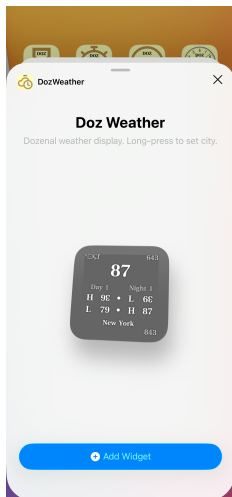
OpenWeatherMap provides current temperature. DozWeather gets forecast information from **Open-Meteo**, but this does not require any subscription or API key.

DOZWEATHER WIDGET

DozWeather includes a Home screen widget, which provides the main functionality: To display weather temperatures in dozenal, using Primel units.

To install the widget, use the same procedure as the other widgets above: Press and hold on the background of a Home screen, until edit mode is triggered. Press the **Edit** button to bring

up the menu. Select the **Add Widget** menu item to bring up the **Search Widgets** dialog. Type “DozWeather” to find the **DozWeather** widget listing and select the icon to bring up the **Add Widget** dialog:



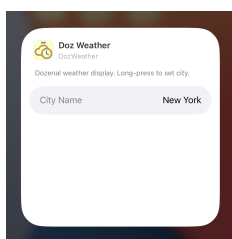
Press **Add Widget** to add it to the Home screen:



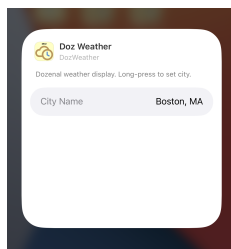
Note that by default the widget is configured to show weather for **New York** city. To change this, press and hold on the widget until the widget menu pops up:



Select **Edit Widget** to bring up the **DozWeather** dialog:

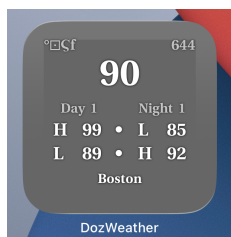


Edit the city name to your own preference, for instance:



Then press the background to dismiss the dialog. The widget now displays temperatures for your chosen city.

Let’s take a closer look at the widget now:



The large numeral in the top center is the current temperature. Beneath it is a table of forecast temperatures. By default, these are all expressed in units from the **Primel** system. Specifically, **degrees-prime-stadial-familiar**, also known as **prime-stadial-degrees** on the **Familiar** scale.

This is indicated by the abbreviation $^{\circ}\square\zeta_f$ in the upper left corner. This symbol includes the brand mark $\square$ for **Primel**, the Greek letter *stigma* (ζ) standing for **stadial**, and the letter *f* standing for the **Familiar** scale. Placing this symbol here acts as a blanket default applicable to all the temperatures shown on the widget.

Since Primel is a dozenal system of measure, all the numerals are implied to be dozenal, without need for annotation. In fact, this also extends to the three-digit numeral in the upper right, which is the local time expressed in dozenal trices (prime-triqua-timels).

The size of the **degree-prime-stadial** was derived by establishing a coherent relationship between energy (in the form of heat) and temperature, via a thermodynamic property of a common substance (water); as well as by recognizing the fundamental equivalence between heat and mechanical work as forms of energy. For any sample of water, the heat needed to raise it by one degree-prime-stadial is the same amount of energy as the work needed to lift it against gravity

by one prime-stadial-height (558_d feet)—about the size of an ancient Olympic stadium! Heat is indeed a much more concentrated form of energy than work.

The thermodynamic property in question is known as “specific thermal capacity,” which Primel terms “massic heatability.” For water, the value of this property varies slightly across its liquid range. By judiciously selecting a standard value within this natural range, we can contrive for the **degree-prime-stadial** to come out to *exactly* $\frac{5}{7}$ Fahrenheit degree.

Unlike other temperature scales, we did not set out to derive this unit by picking two arbitrary anchor temperatures and then dividing the span between them by some “convenient” number. Nevertheless, it turns out that the span between the freezing and boiling points of water comes out to a dozenally-round 190_z (252_d) **degrees-prime-stadial**. This somewhat resembles how the Fahrenheit scale covers this the same span with a temperature difference of exactly 180_d degrees Fahrenheit (between 32_d°F and 212_d°F).

Moreover, by sheer coincidence, a difference of one hundred degrees Fahrenheit is almost exactly the same as one gross degrees-prime-stadial. The Familiar scale takes advantage of this to emulate, in dozenal, what the Fahrenheit scale does in decimal: The span from zero to a hundred degrees Fahrenheit coincides approximately with the span from zero to a gross degrees-prime-stadial-familiar, covering the vast majority of weather temperatures encountered in temperate climates.

Just as the Fahrenheit scale places the freezing point of water about a third of the way between zero and a hundred, the prime-stadial-familiar scale places it *exactly* a third of the way between zero and a gross: 32_d°F = 40_z°□□ ζ_f (exactly).

About two-thirds of the way on both these scales is a reasonable room temperature, if a bit cool: 67_d°F = 81_z°□□ ζ_f (exactly). A little above this is what many find ideal room temperature: 72_d°F = 88_z°□□ ζ_f (exactly).

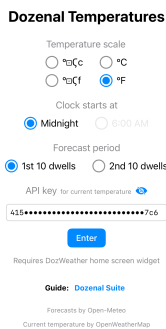
Near the top of this scale is human body temperature: 98.6_d°F = 89.3_z°□□ ζ_f (approximately). In fact with a moderate fever we reach a seeming equi-point: 102_d°F = 102_z°□□ ζ_f (exactly); however, this is just visually similar, since we have one-hundred-two on one scale versus one-gross-two on the other. Either way, as far as human beings are con-

cerned, temperatures that high are certainly becoming *gross*.

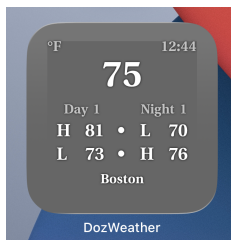
Even the boiling point bears a remarkable resemblance: $212_d^\circ\text{F} = 210_z^\circ\text{C}_f$ (exactly).

You can gain some intuition about this if you interpret a Familiar temperature as a pergrossage, and the equivalent Fahrenheit temperature as a percentage. As such, they'll represent approximately the same fraction of a gross or a hundred as the case may be.

However, DozWeather lets you choose your temperature scale, so you can see these relationships for yourself. Press on the widget or the app icon to bring back the configuration page. Then, under the **Temperature scale** section,



Swipe away the app to see the Home screen again. The DozWeather widget now shows equivalent Fahrenheit temperatures in decimal:



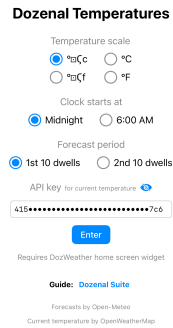
As you can see, 90_z°C_f ($\frac{3}{4}$ of a gross) is approximately 75_d°F ($\frac{3}{4}$ of a hundred).

Note that the time is also being displayed in decimal, in traditional 24_d -hour time: $12:44_d$, i.e. $12:44_d$ P.M. ($12:44_d$ A.M. would be displayed as $00:44_d$.)

For users of Metric, Primel also provides a temperature scale that is more analogous to Celsius, in that the freezing point of ice is set to zero. This is called the **Crystallic** scale (because ice is a crystal), measured in **degrees-prime-stadial-crystallic**, abbreviated $^\circ\text{C}_c$.

To see this, once again press the widget icon to bring up the app configuration

page, then select $^\circ\text{C}_c$ in the **Temperature scale** section:



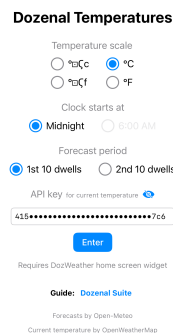
Swipe away the app page to return to the Home screen. The DozWeather widget now shows equivalent degrees-prime-crystallic temperatures, in dozenal:



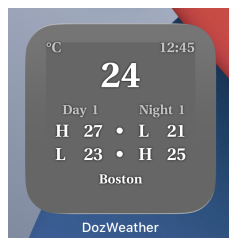
(Note that the current local time is once again displayed in dozenal trices.)

The Crystallic temperatures now all read exactly 40_z degrees-prime-stadial lower than the Familiar temperatures.

To compare these to Celsius, repeat this process but select $^\circ\text{C}$ in the **Temperature scale** section:



Return to the Home screen, and you will see equivalent Celsius temperatures. Note that the current local time is once again displayed in 24_d -hour decimal time.

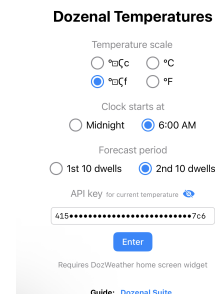


Although the Crystallic scale is analogous to Celsius in setting zero to the freezing point, it's difficult to come up with a way of intuiting Crystallic temperatures given experience with Celsius. Perhaps Crystallic temperatures look like they're "vaguely double" what the Celsius temperatures would be (keeping in mind they're in different bases, so the comparison is only textual). But this seems problematic as the numbers don't always track to a doubling.

The table in the lower half of the widget contains forecast low (L) and high (H) temperatures spanning a pair of adjacent semi-days: a daylight semi-day (labeled **Day 1**) from 3 dwells (average sunrise) to 9 dwells (average sunset) and a night-time semi-day (labeled **Night 1**), from 9 dwells to 3 dwells the following day. Since the time shown above occurs during a daylight semi-day, **Day 1** appears in the left-hand column, and **Night 1** in the right-hand column.

Note that this table is literally a *forecast*, meaning it only reports predicted *future* temperatures. So as the current semi-day wanes, it becomes increasingly likely that the current temperature will end up taking the role of the high or low forecast temperature (or both) in the left column, for whatever is left of the current semi-day.

Back in the app, the **Forecast period** section specifies how far out into the future the highs and lows are being projected:

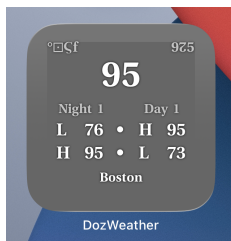


By default **1st 10_z dwells** is selected, giving temperatures forecast for the current semi-day and the next. Selecting **2nd 10_z dwells** gives the forecast for the two adjacent semi-days starting one day from now.

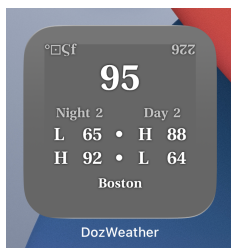


Here, **Day 2** in the left-hand column represents the forecast for the daylight semi-day tomorrow. **Night 2** represents the forecast for the nighttime semi-day tomorrow, going into the day after.

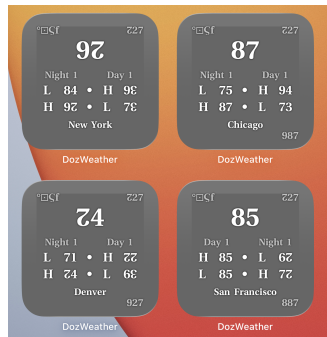
On the other hand, if we wait until the current time is in a nighttime semi-day, **Night 1** will appear in the left-hand column, and **Day 1** in the right-hand column (representing the forecast for the daylight semi-day tomorrow):



If we switch to **2nd 10_z dwells**, **Night 2** will appear in the left-hand column, representing the forecast for the nighttime semi-day tomorrow, going into the day after. **Day 2** will appear in the right-hand column, representing the forecast for the daylight semi-day on the day after tomorrow:



It is possible to put multiple copies of the widget on a Home screen page, and then configure each for a different city, even in other time zones:



Notice that if the given city is in a different time zone, the time in that city appears in the lower right corner, near the name of the city. Bear in mind that time zones generally come in half-dwell increments (60_z trices apart). So for instance, for someone in Boston (Eastern Time zone), the time in Chicago (Central Time Zone) will be 060_z trices earlier, the time in Denver (Mountain Time Zone) will be 100_z trices earlier, and the time in San Francisco (Pacific Time Zone) will be 160_z trices earlier. Since New York is in the same time zone as Boston, no city-specific time is shown in the lower right.

If you go into the app and select °F, all the widgets will be affected. Now all the times will appear in 24_d-hour decimal format:



Unfortunately, there is no way to arrange multiple widgets with the same city but showing different temperature scales.

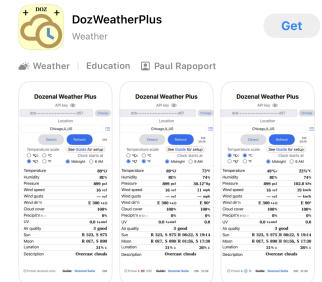
DozClock widgets will update themselves once every 16_z trices (15_d minutes), incurring an API call. This is 8 times per dwell, or 80_z (96_d) per day. Other actions such as closing and opening the DozClock app trigger an API call. This does tap into the 1,000_d free API calls per day, but is unlikely to exhaust it. Extraordinary usage like widgets for multiple cities might start getting more significant, though.

For a complete explanation of Primel temperature units and how they were derived, see the section entitled **Warming up to Temperature** in

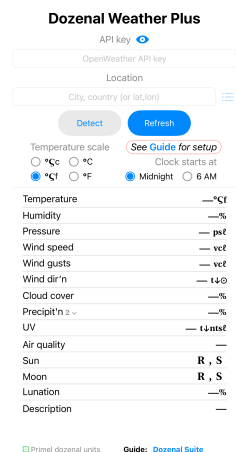
the introductory Primel article, starting on page 19_z (<https://dozenal.org/article-volan-primel-metrology.pdf#page=21>).

DOZWEATHERPLUS

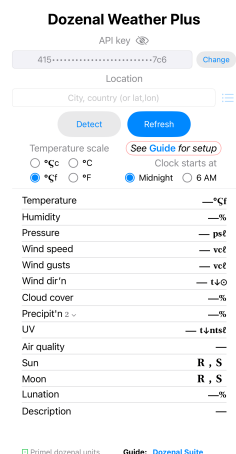
A second, more comprehensive weather app is also available. Search for “DozWeatherPlus” at the App Store to find this entry:



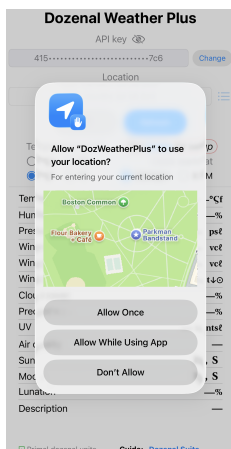
Install this app and open it to land on this screen:



Starting off, it’s pretty inactive. First, copy your API key from DozWeather into the **API key** box. It will need to be from an account at **Open Weather Map** with a **One-Call 4.0** subscription. Once you’ve entered it, make sure to press the eye icon for security:

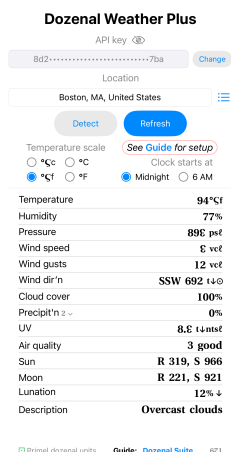


Then press the **Detect** button. A dialog will pop up asking permission to use your location:



Since you will likely want to use **Detect** again in other locations you frequent, choose the **Allow While Using App** option.

DozWeatherPlus will then take your current geolocation and prompt Open Weather Map to identify the city you are in. The end result is something like this:



The **Location** box is filled in with your city, state, and country, and then one-dozen-two (fourteen) pieces of weather data are populated. (See further down for details about each item.)

By default, the values are all in dozenal, using units of measure from Primel. This is indicated at the bottom left corner of the screen, with the blanket default: **Primel dozenal units**.

In the lower right corner is the current time, expressed as a three-digit dozenal number of trices.

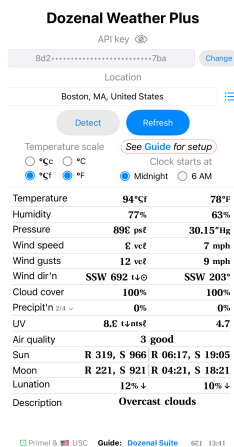
In the **Temperature scale** section, by default the $^{\circ}\zeta_f$ radio button is selected. This means the **Temperature** reading is

using **degrees-prime-stadial** on the **Familiar** scale. Note that, fully qualified for the metrology brand, the abbreviation would be $^{\circ}\zeta_f$. However, because of the global default declared in the bottom left, the brand mark ζ may be omitted to save space.

The radio buttons here control not only the temperature scale, but also what system(s) of measure all the other data are expressed in:

- $^{\circ}\zeta_f$ provides **Primel dozenal units**.
- $^{\circ}\zeta_c$ provides **Primel dozenal units**.
- $^{\circ}F$ provides **USC decimal units**.
- $^{\circ}C$ provides **SI decimal units**.

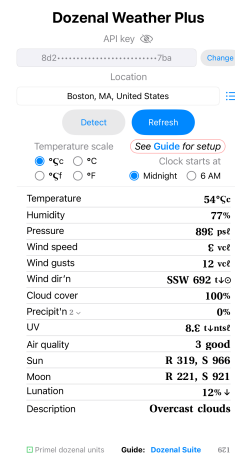
Radio buttons in different rows are mutually exclusive, but radio buttons in the same row may be selected simultaneously. To see this, select the $^{\circ}F$ radio button, while the $^{\circ}\zeta_f$ radio button is already selected:



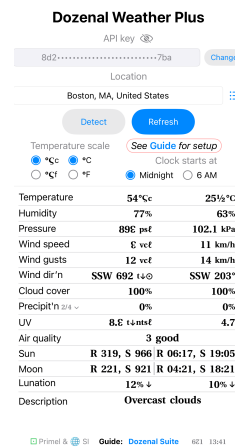
The result is two columns of data: The left-hand column provides dozenal values using Primel units. The right-hand column provides decimal values using United States Customary (USC) units. This is indicated at the bottom left corner as **Primel & USC**.

Now the current time displayed in the bottom right corner is expressed in both dozenal trices and decimal 24_d -hour time. For instance $6\overline{7}1_z$ trices corresponds to approximately $13:41_d$, or $1:41_d$ P.M.

Now select the $^{\circ}\zeta_c$ radio button. The result is one column of data again, with **Primel dozenal units**, but now the **Temperature** reading is on the **Crystalline** scale. The time stamp is once again expressed as dozenal trices only:



Now with the $^{\circ}\zeta_c$ radio button already selected, select the $^{\circ}C$ radio button:

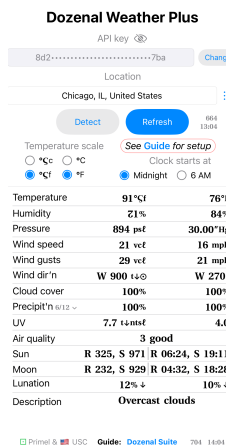


The result is again two columns of data. The left-hand column once again provides dozenal values using Primel units. But now the right-hand column provides decimal values using the International System of Units (SI). This is indicated at the bottom left corner as **Primel & SI**. The timestamp in the bottom right corner is again expressed in both dozenal trices and decimal 24_d -hour time.

One long-time criticism of Celsius is that its degrees are “too big.” They don’t provide a satisfying level of granularity compared to Fahrenheit (and stadial) degrees without adding a fractional digit. DozWeatherPlus alleviates this a bit by rounding to the nearest *half*-degree Celsius, using the vulgar fraction $\frac{1}{2}$ if necessary, as you can see in this last example above.

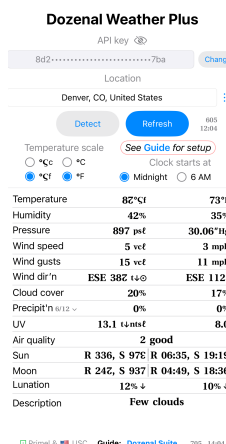
These dual-metrology modes are likely to be the more interesting ones to choose for everyday use. They provide a “Rosetta Stone” like experience, where you can compare Primel readings to equivalent USC or SI values. This can help develop intuition about the Primel units.

You can get weather information for multiple locations. If you travel to another city, you can press the **Detect** button again, and the screen will be populated with the information for your location. You can even type in the name of another city, for instance:

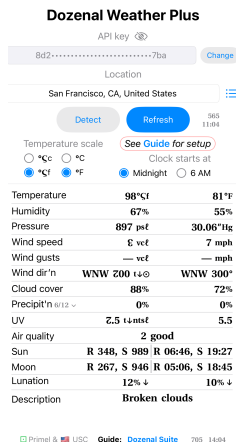


If the city is in a different time zone than your current location (as is the case here), then an additional time stamp is shown just under and to the right of the **Location** box. That gives you the local time at the set location. The timestamp in the bottom right corner of the screen remains your own local time, so you can compare.

Note that time zones come in half-dwell (60_z -trice) increments. Someone located in Boston in the Eastern Time zone will see the time for Chicago in the Central Time zone as 60_z trices earlier. If we select Denver in the Mountain Time zone:

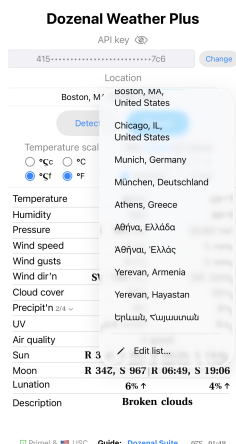


then the time there will appear 100_z trices earlier. Likewise, if we select San Francisco in Pacific Time zone:



then the time there will appear 160_z trices earlier.

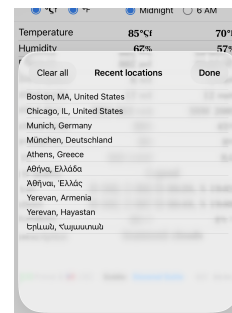
The **Location** field retains a list of the most recent cities you've viewed, which you can access via the "hamburger" icon to the right of the field:



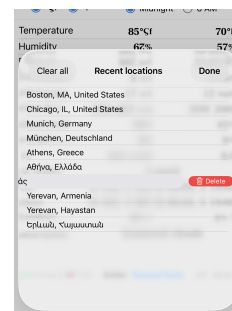
DozWeatherPlus will remember up to a dozen cities this way. More recently viewed cities will evict the least recently viewed ones to keep to this maximum.

Note that **Open Weather Map** is quite flexible about how to identify cities in other countries using languages other than English. As you can see in this example, you can use the English version of a city or country name, or its native-language version. If the English version is an exonym, the service can recognize the endonym as well (for instance **Munich, Germany** versus **München, Deutschland**). You can even use non-Latin script (for instance **Athens, Greece** versus **Αθήνα, Ελλάδα**). It can even recognize a Classical Greek spelling (**Αθήναι, Ἑλλάς**)! You can confirm that these identify the same places by finding no difference in the reported weather data.

At the bottom of the list is the **Edit list...** button. Press this to get a **Recent Locations** dialog:



You can swipe an individual item to the left until an orange **Delete** button appears, and then press that to delete the item from your list. For instance, if you think it's already overkill to use a native script for a foreign city, much less an ancient version of that script, using breath marks and tonal accents that haven't been relevant for a triquennium, you can remove it:



Or you can press **Clear all** to delete the entire list. Press **Done** when you would like to exit this dialog.

You can view a given screen of weather data as long as you like; it will not automatically update. (You can tell this by the timestamp.) Pressing the **Refresh** button makes another call using your API key and updates all the data, as well as the timestamp. Selecting another **Location**, or simply closing and launching the app again, does the same. As long as you don't repeat such actions more than $1,000_d$ ($6\text{E}4_z$) times a day, your subscription will not incur any charges.

Now let's consider all the items of weather data in detail:

- **Temperature:** as already discussed.
- **Humidity:** This gives the moisture in the air as a fraction of the its maximum capacity. In Primel dozenal units, this is a straightforward pergrossage. In decimal USC or SI units, this is a percentage. Both are rounded to the

nearest integer. For instance, 50%_z (five dozen per gross) is approximately 42%_d (forty-two percent).

- **Pressure:** In Primel dozenal units, this is measured in **prime-pressurels**, abbreviated $\square\text{ps}\ell$. Because $\square$ is declared as a default, it can be omitted in the abbreviation $\text{ps}\ell$. In USC decimal units, pressure is measured in **inches of mercury**, abbreviated "Hg. In SI decimal units, this is measured in **kilo-pascals**, abbreviated kPa. Note that 1 $\square\text{ps}\ell$ is approximately 80.35_d Pa or 0.023727_d "Hg.

Average atmospheric pressure at sea level is approximately 891_z $\square\text{ps}\ell$, or 101.325_d kPa, or 29.92" Hg. A pressure above this is considered a "High," which predicts fair weather. A pressure below this is considered a "Low," which predicts foul weather.

- **Wind speed:** In Primel dozenal units, this is measured in **prime-velocitel**s, abbreviated $\square\text{vcl}$. Because $\square$ is declared as a default, it can be omitted in the abbreviation vcl . In USC decimal units, this is measured in **miles per hour**, abbreviated mph. In SI decimal units, this is measured in **kilometers per hour**, abbreviated km/h.

Note that 1 $\square\text{vcl}$ is very nearly 1 km/h (it's 1.0204704_d km/h). So Primel speeds will resemble Metric speeds, except for conversion to dozenal.

(Realize that km/h is not officially an SI unit, because the hour is not an SI unit. The official SI unit for speed is the **meter per second**, abbreviated m/s. But the **prime-velocitel** is Primel's coherent unit of velocity.)

- **Wind gusts:** This represents transient peak wind speed. Same units as for **Wind speed**.
- **Wind direction:** In Primel dozenal units, this is measured in **tricia-turns**, abbreviated $\text{t}\downarrow\odot$. These direction values resemble the time values on a dozenal diurnal clock (with 0 oriented upward), with the dwell hand pointing in the indicated direction. In USC or SI decimal units, this direction is measured in sexagesimal degrees, abbreviated ($^\circ$).

In both cases, the nearest standard compass point is provided as well. Below is a table of the one-dozen-four (sixteen) traditional compass points and their equivalent tricia-turns and sexagesimal degrees:

N	000 _z $\text{t}\downarrow\odot$	0.0 _d $^\circ$
NNE	090 _z $\text{t}\downarrow\odot$	22.5 _d $^\circ$
NE	160 _z $\text{t}\downarrow\odot$	45.0 _d $^\circ$
ENE	230 _z $\text{t}\downarrow\odot$	67.5 _d $^\circ$
E	300 _z $\text{t}\downarrow\odot$	90.0 _d $^\circ$
ESE	390 _z $\text{t}\downarrow\odot$	112.5 _d $^\circ$
SE	460 _z $\text{t}\downarrow\odot$	135.0 _d $^\circ$
SSE	530 _z $\text{t}\downarrow\odot$	157.5 _d $^\circ$
S	600 _z $\text{t}\downarrow\odot$	180.0 _d $^\circ$
SSW	690 _z $\text{t}\downarrow\odot$	202.5 _d $^\circ$
SW	760 _z $\text{t}\downarrow\odot$	225.0 _d $^\circ$
WSW	830 _z $\text{t}\downarrow\odot$	247.5 _d $^\circ$
W	900 _z $\text{t}\downarrow\odot$	270.0 _d $^\circ$
WNW	990 _z $\text{t}\downarrow\odot$	292.5 _d $^\circ$
NW	ᵹ60 _z $\text{t}\downarrow\odot$	315.0 _d $^\circ$
NNW	ᵹ30 _z $\text{t}\downarrow\odot$	337.5 _d $^\circ$

- **Cloud cover:** This represents the fraction of the sky covered with clouds. In Primel dozenal units, this is a straightforward pergrossage. In decimal USC or SI units, this is a percentage. Both are rounded to the nearest integer. For instance, ᵹ8%_z (ten dozen eight per gross) is approximately 89%_d (eighty-nine percent) cloud coverage.

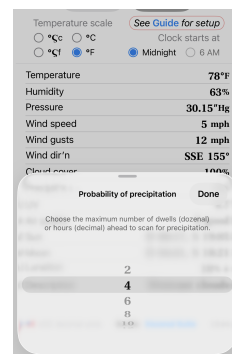
- **Precipitation:** This is the probability of precipitation within a certain amount of time. In Primel dozenal units, this is a straightforward pergrossage. In decimal USC or SI units, this is a percentage. Both are rounded to the nearest integer.

By default this prediction is for 2 dwells (4 hours) in the future. You can see this as a small numeral next to the title **Precipit'n**.

Press this to bring up a menu that lets you choose the number of dwells in the future to display the forecast for, in the range from 1 to 6: In Primel-only modes the setting is expressed in dwells and you may select from 1 to 6 dwells:



In USC-only or SI-only modes, the numeral is expressed in even numbers of decimal hours, and you may select from 2 to 12_d hours:



In dual modes, the numeral is expressed in both:



- **UV:** This is the ultraviolet intensity, equivalent to power density per area. In Primel dozenal units, this is measured in **prime-tricia-intensitel**s, abbreviated $\square\text{t}\downarrow\text{nts}\ell$, equivalent to **prime-tricia-powerels** ($\square\text{t}\downarrow\text{pw}\ell$) per **prime-areanel** ($\square\text{ar}\ell$). Because $\square$ is declared as a default, it can be omitted in the abbreviation $\text{t}\downarrow\text{nts}\ell$. Note that 1 $\square\text{t}\downarrow\text{nts}\ell$ is approximately 1.318_d mW/m². But in decimal USC or SI units, this has been reduced to a "UV Index."
- **Air quality:** This shows the air quality index, providing both a number and a description. This is always one column, and simply reported as given.
- **Sun:** This shows time of sunrise (labeled **R**), and sunset (labeled **S**). In Primel dozenal units, these are in three-digit trices. In decimal USC or SI units, these are specified in 24_d-hour time.
- **Moon:** Similar to the previous, but specifying moonrise and moonset.
- **Lunation:** This represents the fraction of the Moon's disc that is lit up. In Primel dozenal units, this is a straightforward pergrossage. In decimal USC

or SI units, this is a percentage. Both are rounded to the nearest integer. A waxing moon is indicated by an up arrow (↑); a waning moon is indicated by a down arrow (↓).

- **Description** This is a qualitative comment on the weather conditions. This is always one column.

CONCLUSION

There you have it: Seven free dozenal apps for your iPhone.

All activities you engage in with any of these apps reside solely on your device. None of these apps tracks, records, or collects any user data, nor do any sell, lend,

rent, or otherwise transmit or provide any user data.

We hope you find these apps useful, edifying, and enjoyable. We welcome comments, suggestions, and any reports of any bugs you may find. For support with the apps, please contact rapoport.p@gmail.com. For questions about Primel, please contact editor@dozenal.org. ☰

DSA BOARD MEETING

ON SEPTEMBER 5, 1207_z (2026_d), at 730_z trices (8:30_d P.M.) EDT, members of the DSA Board held a meeting (over Zoom). In attendance were:

- GS = DSA President Graham Steele (Member 370_z (552_d)), currently in Kathmandu, Nepal.
- JV = DSA Vice President John Volan (Member 418_z (596_d)), in Ashland, MA.
- MD = DSA Secretary Michael DeVlieger (Member 378_z (527_d)), in St. Louis, MO.
- JS = DSA Treasurer Jay Schiffman (Member 278_z (416_d)), in Philadelphia, PA.
- PR = DSA Board Member Paul Rapoport (Member 230_z (324_d)), in Hamilton, ON, Canada.

Meeting minutes were recorded by MD. JV adapted MD's minutes into this article.

IPHONE APPS DISCUSSION

The meeting was held at PR's request, primarily to discuss his new suite of dozenal apps for Apple iPhone, which he had developed with the help of Claude AI over several months.

(MD and JV had both been testing the apps since May, via the TestFlight app.)

PR initially had some camera problems but soon cleared them up.

On September 4, JV had sent the Board members a draft of an article (see above), which attempts to provide an in-depth pictorial user's manual for PR's app suite.

GS suggested that the *Jottings* mailing should transmit only a link to an HTML landing page for the issue, from which our members can download the PDF at their leisure, rather than send the actual PDF over email. JV reassured him that this was already standard operating procedure. The first issue earlier this year went out exactly that way.

PR highlighted two of his apps that provide temperature and other weather data, including temperature forecasts out to two days.

MD remarked that these apps would be useful in a classroom or other pedagogical setting. He suggested that a follow up app could do length and angle measurement in dozenal. This could be useful for architects, interior designers, workers in the construction trades, etc. Paul replied that to make such an app feasible requires an iPhone Pro with its built-in laser.

JV suggested an idea for an app that could reproduce Apple's Health app, initially the feature to record an outdoor walk, but using Primel units. It would need to access the iPhone's GPS functionality. Distance walked could be represented in Primel "dromals" (almost exactly 2 kilometers each). Time walked could be in Primel "trices." Steps could be counted in dozenal. Pace length could be measured in Primel "morsels" or "ells." Walking speed could be measured in Primel "velocitels."

PR informed the Board that Apple charges iPhone app developers an annual fee of \$112.00_d. He requested the DSA to defray this cost. The Board voted (PR abstaining) and unanimously approved this request. GS pointed out that Apple grants waivers of this fee to non-profit organizations and posted the

link <https://developer.apple.com/help/account/membership/fee-waivers/>. PR agreed to investigate.

JV mentioned that the MailerLite subscription to send out *Jottings* was costing him \$35.00_d per month, but he was treating that as his donation to the DSA. GS suggested finding a cheaper mailing alternative. JV said that when he looked into this at the start of the year, there were cheaper alternatives, but they placed more stringent limits on the number of recipients. However, JV agreed to investigate whether there were better options now.

PLANS FOR ANNUAL MEETING

JV said he had been looking into the possibility of holding an annual membership meeting in Boston in 1207_z (2026_d), like the DSA had done a few years ago. However, the options for suitable meeting rooms with adequate Internet connection and audio/visual equipment looked prohibitively expensive. None of the Board have convenient access to an academic institution like Nassau Community College when Gene Zirkel was still alive.

GS thought that a physical meeting is not as convenient as electronic one, especially for people afield of the meeting site. Why not hold the meeting over Zoom? We could open up participation to the entire membership. MD has a commercial Zoom account, so he can welcome up to a hundred participants.

JV concurred that his experience working in tech during the pandemic impressed him with how much more equitable using Zoom was for all participants. Everyone gets the same experience viewing someone's PowerPoint

presentation on their own laptop. In fact, it got to the point where people in the actual conference room were at a disadvantage squinting up at a projection screen, or trying to be heard on a microphone on the opposite side of a table.

GS wondered if the by-laws require the meeting to be physical. JV posted a link to the DSA by-laws: <https://dozenal.org/article-dsa-constitution-and-bylaws>.

GS expressed concern that we pick a time to accommodate East Asian and other non-Western attendees. JV replied that the membership list contains very few of those, but perhaps they're more motivated to attend.

The Board considered dates that would take advantage of school breaks and avoid busy academic finals weeks. At first, January 7_z (10_d) was considered, but JS said he would be traveling to conferences until a couple weeks later. The Board tentatively agreed on:

SATURDAY, JANUARY $1\mathcal{E}_z$, $120\mathcal{E}_z$
(JANUARY 23_d , 2027_d)

MD said, if this is a solid date, then we can start publicizing it. He also suggested that any newcomers who attend the Zoom could be granted immediate membership and vote. Zoom does support online voting.

GS mentioned he had visited Chaturbhuj Temple, in Gwalior, Madhya Pradesh, India (https://en.wikipedia.org/wiki/Chaturbhuj_Temple,_Gwalior), where he had photographed an inscription containing the oldest known use of a zero. He could present that during the general meeting, and perhaps talk about the cultural impact of zeroes, including the concept of *śūnyatā* ("emptiness") from Madhyanna Buddhism.

MD suggested that JV could give the "Primel Zoom" presentation he first did a dozen years ago, which exploited PowerPoint's animated transitions to zoom in and out while exploring the relative

sizes of things in the universe, in terms of dozenal TGM and Primel measurements. JV said he would work on updating the original Powerpoint and "spiff up" the presentation.

MD suggested he could present some interesting dozenal number sequences from the Online Encyclopedia of Integer Sequences (OEIS), which he regularly contributes to. (In a side discussion, Jay asked Mike for information on a specific number theory solution via Wolfram Mathematica, which Mike uses frequently.)

Of course, any *Jottings* readers can suggest their own presentation topics. Contact the DSA at contact@dozenal.org or editor@dozenal.org.

Updates and details will be published through *Jottings* issues as the meeting date approaches.

NEXT BOARD MEETING

The Board agreed to meet again via Zoom on Saturday, October 15_z , 1207_z (October 17_d , 2026_d), at 730_z trices ($8:30_d$ P.M.) EDT. The goal will be to discuss the agenda for the general meeting in January, and sketch ideas for presentations.

GOODBYE "STADEGREES"

In a side discussion, MD critiqued the portmanteau name of Primel's "stadeegrees" temperature unit, as well as its abbreviation $\boxtimes^\circ$.

The weather apps had been presenting temperatures as for example $88\zeta^\circ_f$ (eight-dozen-eight "stadeegrees" on the "Familiar" scale). Both MD and his daughter had found that confusing, because that made it seem like the ζ (Greek letter *stigma*, standing for "stadial") was some new digit character and part of the numeral.

What had happened was that PR had taken advantage of the dispensations to omit the initial Primel brand mark $\boxtimes$ if it

were displayed prominently as a global default, and also to omit explicit base annotations given that Primel is known to be a dozenal metrology, so the global default also implied a global default base. Then, since customary usage does not put whitespace between a numeral and a "degrees" type temperature unit, PR had left out any space before the ζ , leading to the confusion.

JV tried to defend "stadeegree" because it reflected how Primel's temperature unit was derived based on the equivalence of work and heat as forms of energy. It takes the same amount energy to raise the temperature of a sample of water by a "stadeegree" as it does to lift it against gravity by an entire "stadium" unit (a "stadial" height), which is a pure dozenal scaling of Primel's coherent lengthel unit.

MD countered that things should be as simple as possible, and "stadeegree" just adds too much complication.

JV admitted that he always stumbled a little to pronounce a neologism like "stadeegree" out loud and found himself needing to explain the portmanteau as "stadial" + "degree" every time, so what's the point? In fact, it was the same criticism he himself had raised to the portmanteaus in Do-Metric, such as "duor," "temin," and "vic."

After the meeting, JV took this to heart and officially expunged the "stadeegree" portmanteau from the definition of Primel, replacing it with "degrees-stadial", abbreviated $^\circ\zeta$. Over the next week, JV revised the intro article "The Primel Metrology" at <https://dozenal.org/article-volan-primel-metrology>, and worked with PR to revise the iPhone apps and their descriptions in this *Jottings* issue.

ADJOURNMENT

The meeting adjourned at $\mathcal{E}53_z$ trices ($10:52_d$ P.M.). $\boxtimes$

APPEAL TO MEMBERS

IF YOU HAVE a news item of dozenal interest, or an idea for an article, or a fun creative or artistic work, or even just a question about some dozenal topic, anything you think the rest of the DSA membership would appreciate seeing, do not hesitate to contact editor@dozenal.org. Your item could appear in the next *Dozenal Jottings*! $\boxtimes$

WEBSITE UPDATES

SINCE THE LAST ISSUE of *Dozenal Jottings* the following resources have been added to the <https://dozenal.org> website. (Each of these links points to an HTML landing page with an abstract of the document (reproduced below), containing a clickable thumbnail image linking to a PDF document or other online resource.)

- **Article:** [Systematic Dozenal Nomenclature Summary](#)

Author: John Volan (“Kodegadulo”)

Year: 1203_z (2019_d)

Subject: Systematic Dozenal Nomenclature

A handy one-page reference for Systematic Dozenal Nomenclature, that has been included in some issues of the *Duodecimal Bulletin*.

- **Article:** [The Primel Metrology](#)

Author: John Volan (“Kodegadulo”)

Originally Published: 1203_z (2019_d)

Revised: 1207_z (2026_d)

Subject: Primel Metrology

Primel is a coherent, dozenal-metric, day-gravity-water based metrology (system of measurement). It is the first (i.e. “prime”) metrology to make use of “Quantitels,” a set of generic names for coherent units in any metrology, systematically derived by appending an “-el” suffix (short for “element”) onto the names of the physical quantities they measure; the branding prefix “prime-” (abbreviated as brand-mark ☐) makes them units specific to Primel.

Primel is similar to Tom Pendlebury’s [Tim-Grafut-Maz](#) metrology, but immediately diverges from it by defining the prime-timel as a pure dozenal division of the day. This article provides a first introduction to the system, covering its basic units for Newtonian mechanics (time, acceleration, velocity, length, area, volume, density, mass, force, energy, pressure, etc.) as well as temperature and plane angle. It also demonstrates Primel’s use of [Systematic Dozenal Nomenclature](#) to provide scaling prefixes, as well as its granting of “colloquial names” to many of its units.

This article was originally published in *The Duodecimal Bulletin* [Whole Number 73_z \(123_d\)](#). This standalone version was substantially revised by the author.

- **Article:** [Primel Metrology, Part II](#)

Author: John Volan (“Kodegadulo”)

Originally Published: 1206_z (2022_d)

Revised: 1207_z (2026_d)

Subject: Primel Metrology

This is a follow-on to the introductory [Primel Metrology](#) article. This article introduces “Quantitelic” units (reciprocals of Quantitels), and then covers units of angular mechanics. It resolves the issue of how SI and mainstream mathematicians consider angles to be dimensionless quantities, whereas Primel asserts they are physical quantities with true dimensionality. Part of that resolution involves recognizing a quantity

called “radiality,” and its reciprocal, “curvature.” The article describes Primel units for angular velocity, angular acceleration, angular mass (moment of inertia), angular momentum, angular force (torque).

This article was originally published in *The Duodecimal Bulletin* [Whole Number 74_z \(124_d\)](#). This standalone version was substantially revised by the author to incorporate notational changes introduced in [Article Review: Mohr & Phillips 1188_z \(2015_d\)](#) and [Article Review: Mohr et al 1206_z \(2022_d\)](#).

- **Article Review:** [Mohr & Phillips 1188_z \(2015_d\)](#)

Reviewer: John Volan

Review Date: 1207_z (2026_d)

Reviewed Article: [Dimensionless Units in the SI](#), Peter J. Mohr, William D. Phillips (*National Institute of Standards and Technology*), (1188-05-24_z | 2015-05-28_d)

Subject: Angular mechanics and other units in SI (and Primel)

This article review was adapted from a series of posts in the [Article Reviews about Angular Units](#) thread on the DozensOnline forum at tapatalk. The reviewed article considers certain types of quantities that are treated as “dimensionless” by the International System of Units (SI), in particular angular quantities, but makes the argument that these ought to be treated as having first-class dimension, with units that are more than just pure dimensionless numbers. It expands on this to examine the role of true angular dimension in the various quantities of angular mechanics. One important conclusion is that the radian, and not the turn or cycle, is the only possible “coherent” unit of angle.

The article review strengthens the argument by introducing a notational style that allows symbols in formulas of natural law to default to the SI “dimensionless” interpretation, but marks them with diacritics when they incorporate “true-angle” dimensions. It also uses the same diacritics to mark “complete” versions of trigonometric and other transcendental functions that take “true-angle” inputs or outputs; this allows the original unmarked versions of those functions to default to SI’s “pure math” interpretation working on dimensionless quantities. Throughout, the review uses units from the Primel metrology in examples to demonstrate and clarify these principles, contrasting them with the difficulties caused by SI’s units.

This discussion is continued in [Article Review: Mohr et al 1206_z \(2022_d\)](#)

- **Article Review:** [Mohr et al 1206_z \(2022_d\)](#)

Reviewer: John Volan

Review Date: 1207_z (2026_d)

Reviewed Article: [On the dimension of angles and their units](#), Peter J. Mohr & Eric Shirley (*National Institute of Standards and Technology*), William D. Phillips (*Joint Quantum Institute*), Michael Trott (*Wolfram Inc.*), 1206-04-05_z (2022-04-05_d)

Subject: Angular mechanics and other units in SI (and Primel)

This article review was adapted from a series of posts in the [Article Reviews about Angular Units](#) thread on the DozensOnline forum at tapatalk, and continues the discussion begun in [Article Review: Mohr & Phillips 1188_z \(2015_d\)](#). The reviewed article goes into more depth on the implications of granting angular quantities true dimension, as opposed to the dimensionless interpretation under SI. It revisits the notion of “complete” versions of trigonometric and other transcendental functions, and examines how those interact with calculus. It resolves dimensional dilemmas in such operations by showing how a differential with respect to angle can result in a constant of angular dimension being extracted, with a value that turns out to be a radian, which allows the dimensional analysis to balance. The article then applies this calculus technique to various physical applications involving angular quantities and their related transcendental functions, resolving dimensional imbalances caused by SI’s approach.

The article review strengthens the argument by using the diacritical notations introduced in the previous article review to clarify the symbols and formulas. It also expresses the examples using Primel units to more clearly show the dimensionalities of the quantities via Primel’s “Quantitel” style unit names.

- **Resource:** [Day-Gravity-Water Systems Spreadsheet](#)

Author: John “Kodegadulo” Volan

Latest Edition: 1207-04-12_z (2026-04-14_d)

Type: Online

Nature: GoogleSheets spreadsheet

Summary: GoogleSheets spreadsheet for “Day-Gravity-Water” Based Metrologies

This GoogleSheets online spreadsheet computes coherent units (in the form of Quantitels) for numerous systems of measurement, including Primel, that are in most cases based on a “Day-Gravity-Water” model of metrology construction. Makes use of Systematic Numeric Nomenclature and Computere Transdecimal digits to support multiple bases. The Metrology tab contains a Metrology Selector drop-down for selecting a metrology from the list maintained in the Metrologies tab. Based on this selection, and the parameters of the selected metrology from the Metrologies tab, the Metrology tab displays all the coherent units in the metrology, in the form of Quantitels.

Most users are limited to View Only access. Operating the selection drop-downs on this spreadsheet requires edit access. You may save your own modifiable copy to your Google drive, as long as you abide by the [Creative Commons Attribution 4.0 License](#) on the Copyright tab. This means that you may freely edit your copy, and even distribute it to others, as long as all such copies retain this copyright, with attribution to the original author, as well as retain the same license terms, including a link to the Creative Commons license, and also include documentation of anything you may have modified from the original. If you wish to promote your changes for inclusion into the latest edition, join the [Day Gravity Water Spreadsheet](#) thread at the [DozensOnline](#) forum on Tapatalk. 🍷

NEW MEMBERS

SINCE THE LAST ISSUE of *Dozenal Jottings* the following new members have joined the Dozenal Society of America. If you joined the DSA since our previous issue, look for your name below and note your member number. Regular members are shown in black. Members with *Bulletin* subscriptions are shown in red.

£68_z (1,664_d) Luna L

£69_z (1,665_d) Matthew Kohl Mattes

£67_z (1,666_d) “John Doe”

£68_z (1,667_d) Michael Babcock

£70_z (1,668_d) Henry Worthing

£71_z (1,669_d) John Thomas

£72_z (1,670_d) Isaac Tze Hao Lim

£73_z (1,671_d) Jonathan Taylor Martin

£74_z (1,672_d) Brett S Fudge

£76_z (1,674_d) Alexander Schrod

£77_z (1,675_d) Rachel J Nguyen

£78_z (1,676_d) Alastor Drinkwine

£79_z (1,677_d) Monty Craig

£77_z (1,678_d) Orhan Demir

£78_z (1,679_d) Victor Aguilar

£80_z (1,680_d) Zach Lym

£81_z (1,681_d) Steve Postma

£82_z (1,682_d) James Cary Stargel

£83_z (1,683_d) Thomas Burke

£84_z (1,684_d) Aaron Firestone

£85_z (1,685_d) Zachary A. Newton

£86_z (1,686_d) Teresa Steen

£87_z (1,687_d) Alexandra Borgzinner

£88_z (1,688_d) Elijah Matthew White

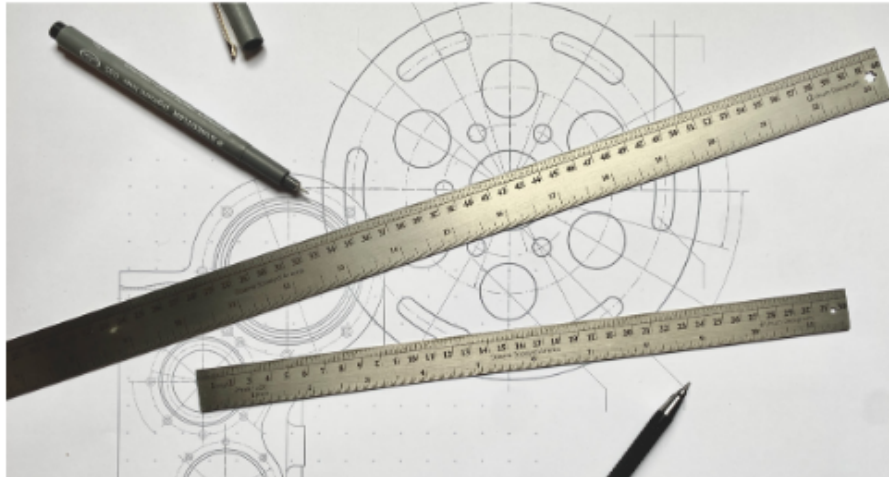
£80_z (1,689_d) Richard Gott

£87_z (1,690_d) Gia Bamrud 🍷

EACH ONE, TEACH ONE

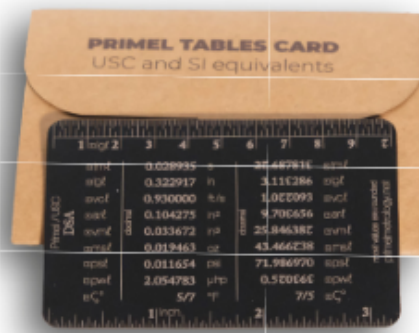
PRECISION DOZENAL TOOLS

Available at unumd.company.site



Primel Rulers

These double-sided rulers display Primel-Lengthths on the top edge, divided into uncias, with a special design to highlight halves, thirds, quarters and sixths. The bottom edge shows Inches (USC) on one side, Centimeters (SI) on the other. Available in two sizes: three dozen and six dozen primel-lengthths (almost exactly equivalent to one and two Grafuts from Tom Pendlebury's TGM system). Made to last: stainless steel, laser engraved, wooden case.



Primel Tables Card

This double-side engraved card features conversions between Primel and USC units on one side, and between Primel and SI units on the other side. Each side has conversions from Primel and to Primel. High-quality aluminum - Credit card format.



Dozenal Tables Card

This double-sided engraved card features a dozenal multiplication table on one side and a dozenal addition table on the other. Carry your tables in your pocket and learn them at your own pace. High-quality aluminum - Credit card format.

Your purchase supports the DSA!

These high-quality products were designed in conjunction with the Dozenal Society of America. 20%(per gross) of Unum Designum's profits will be donated to support the DSA's dozenal outreach and educational work.