What an .ics Calendar Export Actually Encodes

> An exported calendar file has to make explicit choices about time zones and all-day events that a plain date or date-time value on a web page never had to make.

Exporting a deadline or milestone as an .ics file (the standard calendar file format most calendar apps can import) requires resolving two things a page display never has to, whether the event is a specific zoned instant or a "floating," timezone-independent all-day date, and how that choice will be interpreted by whatever calendar application eventually imports the file, which may be running in a different time zone than the one the deadline was originally planned in.

Zoned instant (e.g. a 2:00 PM deadline)

Importer sees 2:00 PM converted to their own zone

All-day date (e.g. a milestone with no specific time)

Importer sees the same calendar date, no time attached

A zoned instant converts for the importer's own time zone; an all-day date is deliberately timezone-independent and shows the same date everywhere.

Why an all-day milestone should not carry a time zone at all

A deadline that is genuinely date-only, such as "submissions due March 15" with no specific hour attached, is best represented in .ics as a floating all-day date rather than a specific zoned instant, precisely so that it displays as March 15 in every importing calendar regardless of what zone that calendar is set to. Encoding it instead as, say, midnight in a specific zone risks the imported event showing on the wrong calendar date for someone in a sufficiently different zone.

Why a specific-time deadline needs the opposite treatment

A deadline with a genuine specific time, such as "submissions close at 5 PM Eastern," should be encoded as a zoned instant so that an importer in a different zone sees it correctly converted to their own local time, preserving the actual real-world moment the deadline falls at rather than a literal, un-converted clock reading that would be wrong for them.

What WhenMath's .ics export does

The engines that offer an .ics export (multi-phase deadline planning, review and approval timelines, and rolling deadlines) generate RFC 5545-compliant all-day VEVENT entries for date-only milestones, which is the correct encoding for the kind of plan-level milestones those specific engines produce. A future engine exporting a genuinely time-specific deadline would need the zoned-instant encoding instead, which the underlying .ics writer module also supports.

Key takeaway. A date-only deadline should export as a floating all-day .ics event so it displays on the correct date everywhere. A deadline with a genuine specific time should export as a zoned instant so it converts correctly for the importer's own time zone.

Frequently asked

Why not just always use a zoned instant for everything?

Because a date-only milestone has no real specific time to anchor a zoned instant to, forcing one (typically midnight) risks the event appearing to fall on the wrong calendar date for an importer in a meaningfully different time zone, which a floating all-day date avoids entirely.

Does every calendar app handle floating all-day events the same way?

Mainstream calendar applications generally handle RFC 5545 all-day events consistently, since the format explicitly defines them as date-only with no time-zone component, but WhenMath cannot verify every third-party calendar application's import behavior.

Can I still see the deadline's time on the WhenMath page even if the export is all-day?

Yes, the on-page result and the exported file serve different purposes. The page can show full detail including any relevant time, while the export intentionally simplifies a date-only milestone to just its date for reliable cross-zone display.

WhenMath is a browser-local calculation tool, not a legal, payroll, or scheduling authority. This guide explains a calculation convention; it is educational, not a determination that any specific contract, employer, or jurisdiction follows it.