

APEX Scheduler — Installer & User Guide

Driver: SmartAPEX-Lighting-Scheduler-V1 (shown in Composer as *SmartAPEX APEX Scheduler V1*, in the app as **APEX Scheduler**)

Version covered: 61 · Control4 OS 3.1.3 or newer · September 2026

APEX Scheduler lets the home owner create their own automation from the Control4 app, without Composer:

- **Schedules** — lights, blinds, thermostats, relays and audio rooms at a clock time, sunrise/sunset or prayer time, on chosen weekdays, once on a date, or between two dates.
- **Scenes** — modes such as Bedtime, Away or Movie that the customer defines once, runs by hand, and reuses inside schedules.
- **Daylight** — a group of lights follows a brightness / white-temperature curve from a start to an end time, switching on at the start and off at the end (which Control4's own Daylight agent never does).
- **Music & prayer times** (Pro) — room audio in schedules, and adhan at exact prayer times with per-country calibration.
- **Voice control** — any scene or schedule can be run from Alexa, Google or Josh through a Control4 Voice Scene.
- **Backup** to a USB stick, automatic copies on the controller, and Export/Import in Composer.

Contents

1. Requirements.....	2
2. Installation	2
3. Licence	2
Base licence (hardware).....	2
Pro licence (rooms/music + prayer times)	3
4. Installer setup (Composer)	3
4.1 Presets — music, playlists, announcements, anything from Programming	3
4.2 Adhan (call to prayer) audio	3
4.3 Relays	3
4.4 Thermostats	4
4.5 Blinds and curtains.....	4
4.6 Prayer times (Pro)	4
4.7 Other properties	4
5. Using the page (for the customer).....	4
5.1 Home screen	4
5.2 Creating a schedule.....	5
5.3 Scenes.....	7
5.4 Daylight	8
5.5 Behaviour to know.....	9
6. Music and rooms (Pro).....	9
6a. Voice assistants (Alexa, Google, Josh).....	9

7. Backup and restore	10
8. Composer actions.....	11
9. Troubleshooting.....	11
10. Limits	12

1. Requirements

Item	Requirement
Control4 OS	3.1.3 or newer (WebView UI Button). Tested with Composer Pro 2026.5
Licence	At least one SmartAPEX IP module online in the project: Mini IP Gateway V2 (driver ≥ 987), IP Relay V2 (≥ 1028) or DALI Gateway V1 (≥ 338). Older gateway drivers do not publish their online state — update them first
Pro features	Rooms / music and prayer times need a Pro key (30-day free trial starts automatically)
Sunrise / sunset / prayer	The project's Latitude and Longitude must be set in Composer (Project > Properties). Without them the page says <i>Sunrise/sunset schedules are unavailable</i>

2. Installation

1. Copy `SmartAPEX-Lighting-Scheduler-V1.c4z` to `Documents\Control4\Drivers` and add it from **My Drivers** into any room (a rack or "Utility" room is fine). One instance per project.
2. The device appears in the app under that room as **APEX Scheduler** with its own icon. If an instance from an older build is still called *Lighting Scheduler*, run the action **Rename to APEX Scheduler** once.
3. Open the driver's page in the app. The first line under the title shows the controller clock, sunrise/sunset, Maghrib (when Pro) and the driver version.
4. **Updating the driver:** *Update Driver* in Composer can restart Director for 20–60 s (the driver declares a proxy, icons and events). Do it when nobody is using the system. The page reloads itself once after an update.

3. Licence

Base licence (hardware)

The driver runs only while a SmartAPEX IP module is **really online** — the gateway drivers publish a variable *APEX Online* that the scheduler reads every 5 minutes. Adding a gateway driver without the hardware does not unlock it.

- Property **License** and the **Status** property show the result, e.g. *Licensed via SmartAPEX Dali Gateway V1* or *UNLICENSED – ...; schedules will not run*.
- Several modules: any one of them online is enough; the summary reads *N of M modules online*.
- A short outage is tolerated: 24 hours of grace after the last time a module was seen.

- Action **Check License** re-checks immediately.

Pro licence (rooms/music + prayer times)

- A 30-day trial starts when the driver is first used; the page shows *Pro trial, N days left*.
- The key is bound to the controller. Send SmartAPEX the controller MAC shown in the **Pro Status** property and you receive a key in the form `APEXPRO-<features>-<expiry>-<signature>`.
- Paste it into **Pro License Key**; **Pro Status** confirms. Action **Check Pro License** re-validates.
- Without Pro: room cards and prayer times are shown with a **PRO** badge and refuse to save or fire; everything else keeps working.

4. Installer setup (Composer)

4.1 Presets — music, playlists, announcements, anything from Programming

The driver has 20 events, **Preset 1 ... Preset 20**. A preset is the bridge between the customer's page and things only Programming can do (playlists, media scenes, announcements, third-party gear).

1. **Properties → Preset N Name**: give it the name the customer will see, e.g. *Morning Jazz* or *Adhan*. Presets with an empty name are hidden from the page.
2. **Programming → Events → APEX Scheduler → Preset N**.
3. Add the actions, e.g. *Living room → Audio → Play playlist "Morning Jazz", Set volume 30*. For several rooms add several actions or use a Media Scene.
4. Test with **Execute** in Programming.
5. On the page the customer picks the preset in a room card (**Play** row) or in the *Installer presets* chips of a schedule's on/off time.

Presets are **events**, so they do not appear in Composer's *Media* tab. Media is the controller's own library; the link between a preset and a track is always made in Programming.

4.2 Adhan (call to prayer) audio

Treat the adhan like music, not like an announcement:

1. Put the adhan MP3 in the controller's media library (Composer → Media → My Music, scanned from a USB stick on the controller or a network share). One file per muezzin if wanted.
2. Create a playlist, e.g. **Adhan**.
3. Programming: *APEX Scheduler → Preset N → Room → Audio → Play playlist "Adhan"* (+ volume). Name the preset *Adhan*.
4. The customer creates a schedule: **When → Prayer → Maghrib** (or Fajr...), adds the room card, picks **Play → Adhan**, and optionally *Room off* a few minutes later in the off half.

The Announcements agent is designed for short clips; a 3–5 minute adhan is safer as a playlist.

4.3 Relays

30 relay connections, **Relay 1 ... Relay 30**. Bind any relay output (from a SmartAPEX IP Relay, a controller relay or any relay driver) in *Connections*, then name it in **Properties → Relay N Name** — that name appears on the page. Unbound or unnamed relays are hidden. Actions: on (close), off (open) or pulse (0.1–600 s).

4.4 Thermostats

Every thermostatV2 device in the project is offered. Modes, fan modes and setpoint range are read from the thermostat; SmartAPEX CAN Thermostat V4 uses its own lists and a single setpoint (15–35 °C). **Properties** → **Temperature Scale** (Celsius / Fahrenheit) decides how setpoints are shown and sent; default Celsius.

4.5 Blinds and curtains

Every blind proxy device is offered. Drivers that report a level axis get open / close / position %. Relay-type Control4 blind drivers (open/close only, optional stop) get open / close. SmartAPEX Curtain V1 needs its travel limits stored first, otherwise it refuses positions.

4.6 Prayer times (Pro)

Property	Meaning
Prayer Country	Oman, United Arab Emirates, Qatar, Saudi Arabia, Kuwait, Bahrain, Egypt, Jordan, Iraq, Lebanon or <i>Custom</i> . Selecting a country fills Prayer City with that country's cities only and sets the calculation method and official corrections
Prayer City	City (coordinates) or <i>Project location</i> (Composer latitude/longitude)
Prayer Calculation Method	Umm al-Qura, Egyptian, MWL, ISNA, Karachi, ... (preset by the country, editable)
Asr Calculation	Standard (Shafi'i/Maliki/Hanbali) or Hanafi
Prayer Minute Adjustments	Five numbers in minutes: Fajr, Dhuhr, Asr, Maghrib, Isha, e.g. 0,0,0,0,0. Added on top of the country's built-in corrections

Oman is calibrated against the Ministry (MARA) table for Muscat to within ± 1 minute. Other countries use the standard method of their authority; send SmartAPEX an official table to calibrate a city. Action **Print Prayer Times** prints today's times to the Lua output — compare them with the local table before going live.

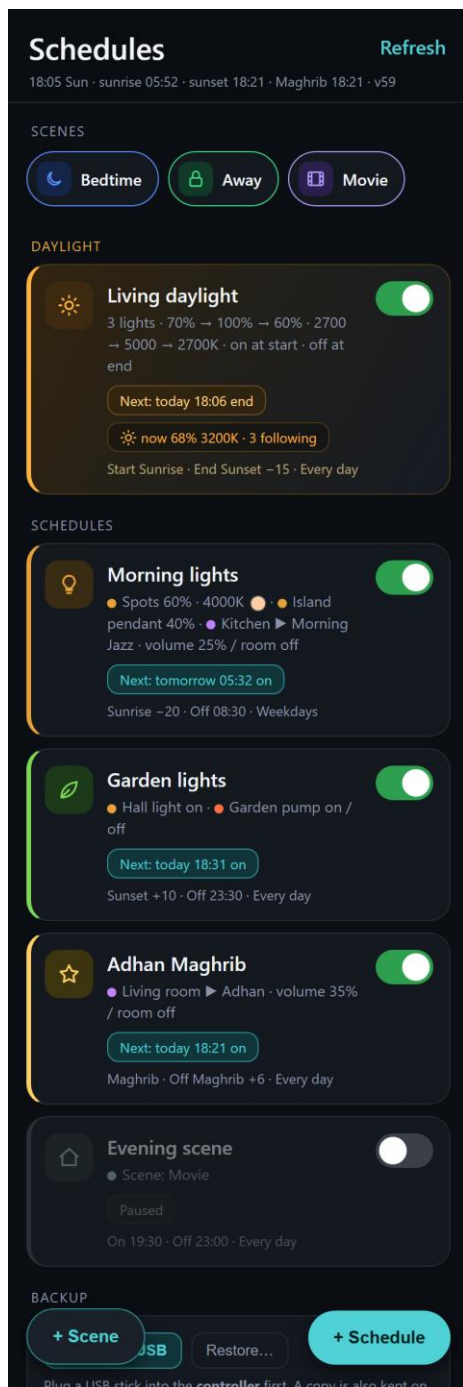
Prayer events fire on the exact second of the minute (the engine arms a one-shot timer for HH:MM:00).

4.7 Other properties

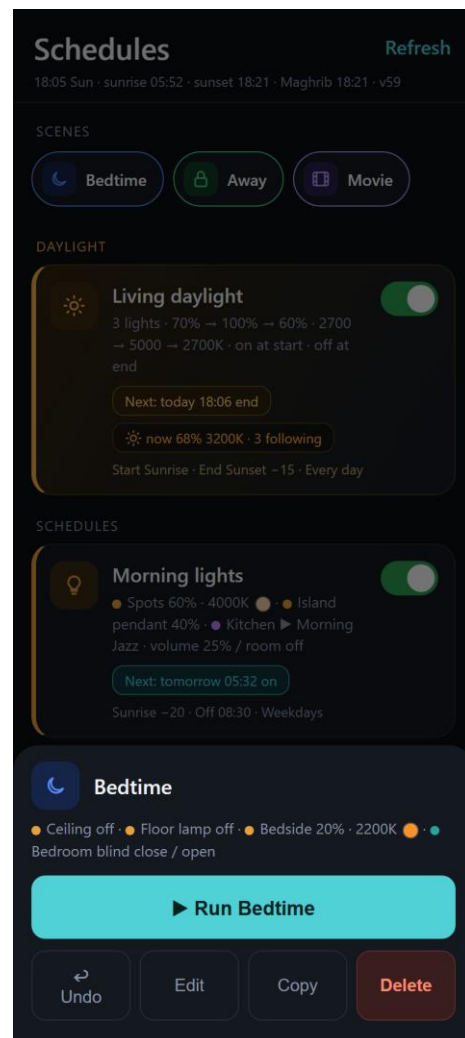
Property	Meaning
Status	Number of schedules, how many are on, and the soonest next event
Web Path	Path of the page inside the driver; leave contents/index.html
Import JSON / Import Mode	See §7
Debug	Verbose Lua output (ticks, arming, sends)
Test Light ID / Test Blind ID	Targets for the <i>Test Light</i> / <i>Test Blind</i> actions

5. Using the page (for the customer)

5.1 Home screen



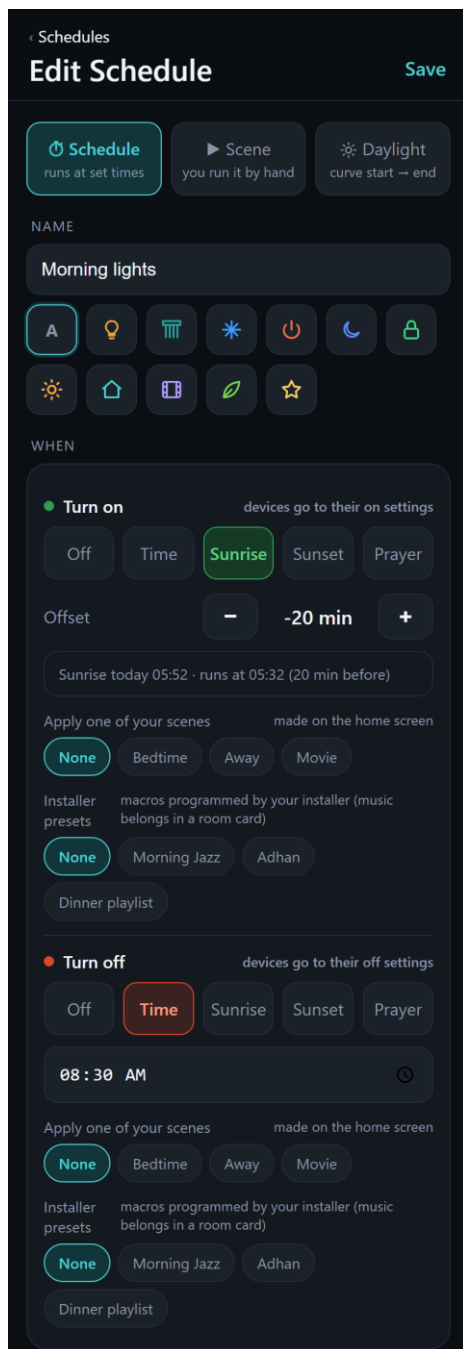
Home screen: scenes, Daylight and schedules



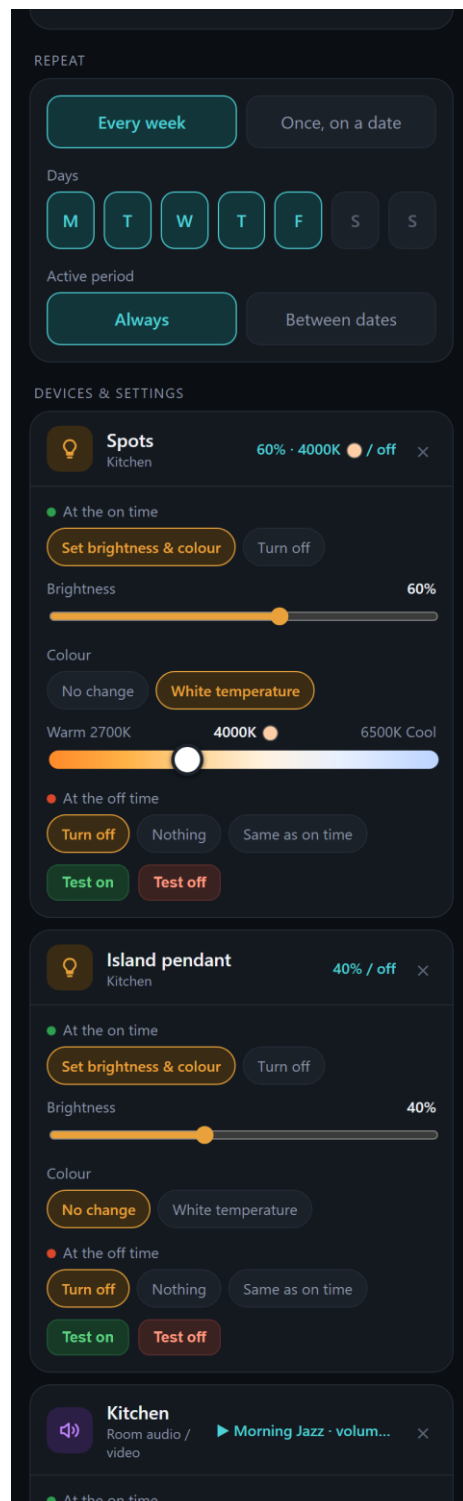
Tapping a scene: Run, Undo, Edit, Copy

- **Scenes** — a row of pills at the top. Tap one: **Run** (applies each device's *on* settings), **Undo** (the *off* settings), **Edit**, **Duplicate**, **Delete**.
- **Daylight** — amber cards, one per curve, with a live line *now 78% 4100K · 2 following*.
- **Schedules** — one card each: devices, next event, trigger and days. The switch on the right pauses a schedule (the card dims). Tap a card for **Edit**, **Duplicate**, **Test on / Test off**, **Delete**.
- **Backup** — see §7. **Diagnostics** shows the page log.
- **+ Schedule** and **+ Scene** at the bottom.

5.2 Creating a schedule



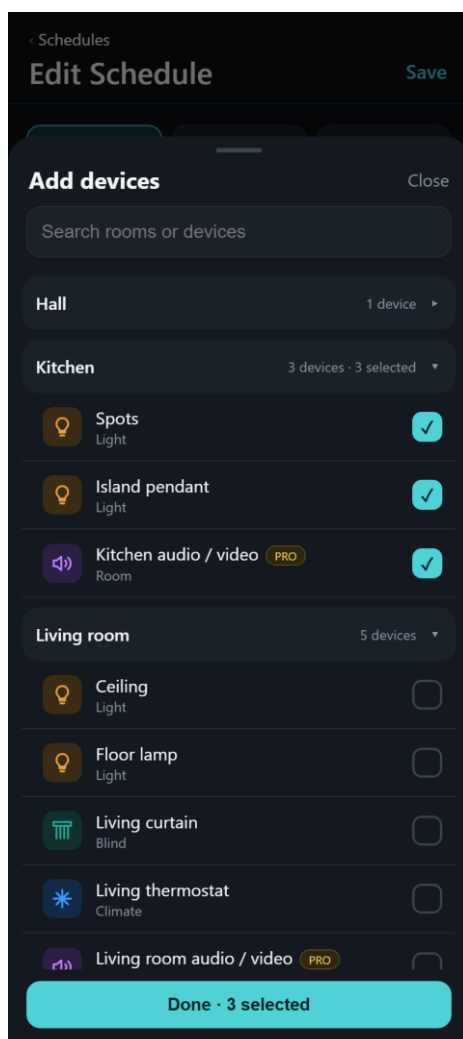
Editor: kind, name, icon and the When panel



Repeat and the device cards (light, room)

1. **Kind:** Schedule (runs at set times) · Scene (you run it by hand) · Daylight (curve start → end).
2. **Name** — leave empty for an automatic name (*Living lights 18:00, Kitchen blinds sunrise ...*). Pick an icon or leave A (automatic).
3. **When** — *Turn on* and *Turn off* halves, each: Off (this half does nothing), **Time**, **Sunrise**, **Sunset** or **Prayer** with an offset in 5-minute steps (–360...+360). The line under it shows today's real time. An offset that would leave the day (e.g. sunset +6 h) is skipped, never wrapped to the next day. Under each half you can also pick one of **your scenes** and one **installer preset**.

4. **Repeat** — *Every week* on ticked days, optionally only *Between dates*; or *Once, on a date* (it switches itself off afterwards).
5. **Devices & settings** — *+ Add device* opens a sheet grouped by room (search box on top). Each device becomes a card; tap the header to fold it.
 - **Light**: at the on time *Set brightness & colour* (slider, white temperature or RGB) or *Turn off*; at the off time *Turn off, Nothing* or *Same as on time*. On/off-only lights show *Turn on / Turn off*. *Test on* sends brightness and colour immediately.
 - **Blind**: open / close / position % (when the driver supports it) for each half.
 - **Climate**: mode (Off / Heat / Cool / Auto / Fan...), setpoint, fan — colour-coded red / blue / green. *No change* leaves a value alone.
 - **Relay**: on / off / pulse.
 - **Room (Pro)**: *Room off, Volume %, Mute / Unmute* and **Play** an installer preset.
6. **Preview** — *Test the on time / Test the off time* run the unsaved schedule now. **Ramp** (seconds) applies to lights.
7. **Save**.

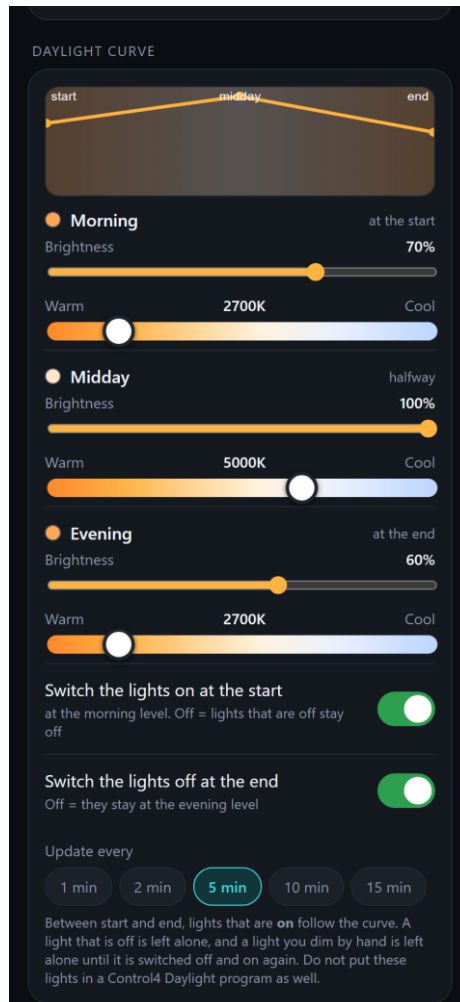


Add device sheet, grouped by room

5.3 Scenes

A scene is a schedule without times. Create it with *+ Scene* (or switch the kind in the editor), set each device's *run* and *undo* settings, save. It appears in the Scenes row and inside every schedule's When panel as *Apply one of your scenes* — e.g. a *Bedtime* scene applied by a 23:00 schedule and also runnable by hand. A scene can reference another scene one level deep.

5.4 Daylight



Daylight editor: the curve, the two switches and the step

1. Kind **Daylight**. Start and End: time, sunrise/sunset or prayer, with offsets. End must be after start (same day).
2. **Curve**: Morning (at the start), Midday (halfway), Evening (at the end) — brightness and white temperature each; the graph previews it.
3. **Switch the lights on at the start** / **Switch the lights off at the end** — the two things Control4's Daylight never does. Both can be off if you only want lights that are already on to follow.
4. **Update every** 1 / 2 / 5 / 10 / 15 minutes (5 is a good default; 1 minute with 40+ lights loads a DALI bus unnecessarily).
5. Add lights. All lights in one Daylight follow the same curve; make a second Daylight for a different curve. **A light can be in only one Daylight** — the page refuses otherwise.

Rules while the curve runs:

- A light that is **off** is left alone (same rule as Control4). A light switched on later joins the curve at the next step.
- A light you **dim by hand** is left alone until it is switched off and on again; the card says *N by hand*.
- White temperature goes to lights with a CCT channel; RGB-only lights mix it from RGB; on/off lights only take part at the start and the end.
- Do **not** put the same lights in a Control4 Daylight program as well — the two would fight.
- *Apply now* on the card applies today's curve value immediately (switching the lights on); *Lights off* switches them off.

5.5 Behaviour to know

- Events fire on the exact second of the minute.
- After a Director restart or driver update, events of the last 15 minutes are caught up (logged as LATE); older ones are skipped.
- Changing a schedule while its event is armed re-checks it at fire time; a moved time does not fire twice.
- When the licence is lost, nothing fires; the page shows a red banner.
- Lights get a *settle* pass: after the ramp the level is re-sent and the colour applied, so a light coming from off lands on the exact value instead of its own On preset.

6. Music and rooms (Pro)

A room card offers *Room off*, *Volume*, *Mute/Unmute* and **Play** → **preset**. Volume and Play together mean: the preset starts the music (through Programming) and the volume is set. Playlists cannot be selected from the page directly — Control4 exposes no public API for that — hence the presets in §4.1.

6a. Voice assistants (Alexa, Google, Josh)

Alexa and Google only see what Control4 exposes: lights, Lighting Scenes and **Voice Scenes**. The bridge is a Voice Scene that runs one of the customer's APEX scenes:

1. Composer → Agents → **Voice Scenes** → add a scene, e.g. *Bedtime*.
2. Programming → Events → Voice Scenes → *Bedtime* **activated** → Actions → APEX Scheduler (device commands) → **Run Scene**: pick *Bedtime* from the list, Action *Run*. Optionally *deactivated* → the same with *Undo*.
3. In the Alexa / Google app run device discovery; *Bedtime* appears as a scene. "Alexa, turn on Bedtime" now runs the customer's scene.

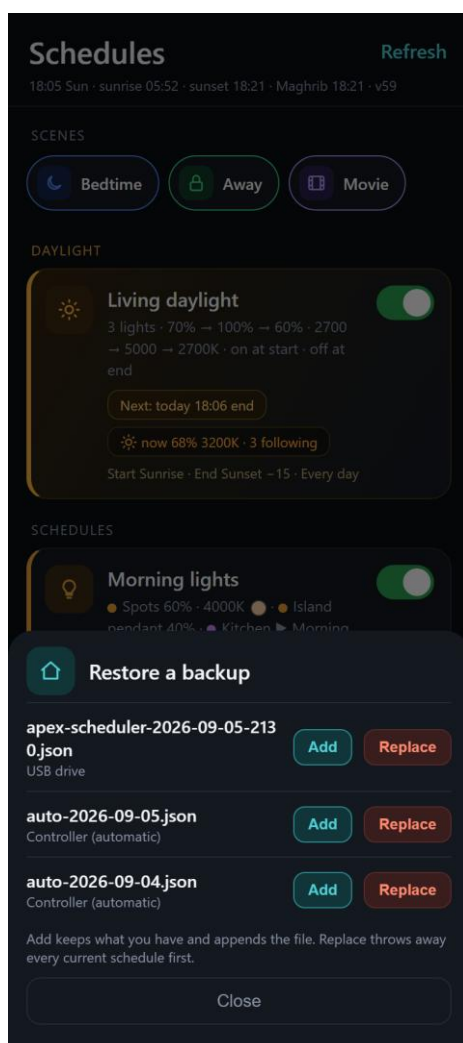
Notes:

- **Run Scene** lists the current scenes and schedules (scenes first; schedules and daylights are marked). **Run Scene By Name** takes free text instead — useful when the scene will be created later; matching is case-insensitive and a scene wins over a schedule of the same name.
- A schedule can be run the same way (its *on* half = Run, *off* half = Undo); a Daylight run applies the current curve value and switches its lights on.

- Each new scene the customer wants by voice needs one Voice Scene wired by the installer. Prepare a handful with common names (Bedtime, Away, Movie, Morning, Dinner) and tell the customer to use those names.
- If Programming names a scene that does not exist, nothing happens and the page shows *Programming asked for '...' but no scene has that name.*
- **If the Run Scene list does not open** in your Composer version (the CUSTOM_SELECT dialog is an OS 3.3+ feature and has not been verified on every OS 3.x build), use **Run Scene By Name** instead — it does exactly the same with the name typed in. Both commands are under *APEX Scheduler → Device Specific Commands* in the Programming actions pane; if neither shows, run *Update Driver* once and reopen Programming.

7. Backup and restore

- **Backup to USB** (page): plug a USB stick (FAT32) into the **controller**, tap the button. The driver writes `apex-scheduler-YYYY-MM-DD-HHMM.json` into a folder `apex-scheduler` on the stick. Without a stick the page says *No USB drive found. Plug a USB stick into the controller (not the phone) and try again.*
- **Restore...** (page): lists the USB copies and the automatic copies; **Add** appends, **Replace** (tap twice) discards the current table first.



Restore menu: USB copies first, then the automatic copies

- **Automatic copies:** after every change the driver rewrites `auto-YYYY-MM-DD.json` in its own folder on the controller and keeps the last 7 days.
- **Composer:** action **Export Schedules** prints one JSON line to the Lua output; paste it into **Import JSON**, choose **Import Mode** (Add / Replace) and run **Import Schedules**. Ids are kept when free, scene references are remapped, the property empties itself afterwards.
- The Composer project backup also contains the table (PersistData).
- Moving to another project: device ids differ, so re-select the devices in each schedule after importing.

The backup is the customer's data only (names, times, device ids, levels); no driver code or keys.

8. Composer actions

Action	What it does
Rename to APEX Scheduler	Renames the proxy device (brief Navigator refresh)
(Programming commands) Run Scene / Run Scene By Name	Run or undo a scene or schedule from Programming — see §6a
Check License / Check Pro License	Re-check now, result in the properties
Print Schedules	Every row as JSON + today's sun times
Print Daylight State	Window, curve, per-light level, <i>by hand</i> flags
Print Prayer Times	Today's times for the configured city
Print Thermostats / Print Rooms / Print Relays / Probe Lights	What the driver found and each device's capabilities
Export Schedules / Import Schedules	See §7
Backup to USB / Print USB Drives	Write a backup; list what the controller sees under /media
Run Tick Now	Force one engine pass
Test Light – On 60% / Off · Test Colour – All Routes · Test Blind – All Variants	Bench probes for one device (ids from the Test properties)
Reload Web URL	Re-announce the page URL to Navigator

9. Troubleshooting

Symptom	Check
Red banner <i>Not licensed</i>	A SmartAPEX IP module must be online and its driver new enough to publish <i>APEX Online</i> (Mini IP ≥ 987, Relay ≥ 1028, DALI ≥ 338). Run <i>Check License</i> ; the License property names the reason
Sunrise/Sunset/Prayer greyed	Set Latitude/Longitude in Composer project properties, then <i>Refresh</i>
Prayer chips show ★ / refuse	Pro trial ended — enter the Pro key

Symptom	Check
Prayer time differs from the local table	Compare <i>Print Prayer Times</i> with the table; adjust Prayer Minute Adjustments or send the table to SmartAPEX for calibration
Setpoints look wrong (°F)	Temperature Scale property
Daylight: lights do not follow	<i>Print Daylight State</i> — <i>level now</i> must be a number. A light whose level cannot be read (variable 1001) is never driven; a light at 0 is off and left alone by design
Daylight: a light stopped following	It was changed by hand; switch it off and on
Two lights "fight"	The light is also in a Control4 Daylight program — remove it from one
Backup: <i>No USB drive found</i>	Stick must be in the controller, FAT32. <i>Print USB Drives</i> lists what <code>/media</code> contains
Page shows an old layout after an update	It reloads once by itself; otherwise close and reopen the page
Preset does nothing	Empty Preset N Name hides it; check the Programming action with <i>Execute</i>
Alexa scene does nothing	Voice Scene must be wired to <i>Run Scene</i> in Programming; the scene name must exist (see the Lua output / page toast)
Blind position slider missing	The blind driver reports no level axis (relay-type) — open/close only
Curtain refuses a position	SmartAPEX Curtain V1 needs its travel limits stored

10. Limits

- One driver instance per project, 20 presets, 30 relays; no limit on schedules other than good sense (the page is tested with 20+).
- Times are the controller's local time; sunrise/sunset ± 2 min (NOAA method).
- Daylight midday is halfway between start and end, not solar noon.
- Verified on SmartAPEX DALI Light V1, SmartAPEX Light V2, SmartAPEX Curtain V1, SmartAPEX CAN Thermostat V4, Control4 relay/blind drivers and generic light_v2 dimmers. Other drivers should work through the standard proxies; report anything that does not.
- Not yet verified on hardware at the time of writing: USB mount name under `/media` on every controller model (the driver tries every entry), Composer's maximum length for the *Import JSON* property (use Add mode in parts if it truncates), and the *Run Scene* pick-list dialog in Programming on OS 3.x (fallback: *Run Scene By Name*).

SmartAPEX · smartapex.app