

Cuelio Studio Rugby Playout

User guide and template field reference

Version 0.1.0 · 14 October 2026

CasparCG playout control client for live rugby broadcasts

Contents

1. Getting started
2. The panel at a glance
3. Before the match: Event Info, Teams and Tables
4. During the match: scoring, clock and graphics
5. The playout grid and keyboard
6. Settings and safety nets
7. Chroma mode: playing out from one HDMI port
8. For template designers: how graphics receive data
9. Template field reference (every graphic, every field ID)
10. Troubleshooting

1. Getting started

Cuelio Studio Rugby Playout drives a CasparCG server over AMCP and keeps the match itself: score, clock, period, cards, possession and set pieces. Pressing **Try** both changes the score and refreshes any graphic on air that shows it.

Starting the app

- **Desktop app:** start Cuelio Studio Rugby Playout from its shortcut. Only one copy runs at a time; starting it again brings the open window forward.
- **Second screen:** while the app runs, any computer on the same network can open the panel in a browser at `http://<this computer>:7200`.

Licence

A new installation runs a **7-day free trial** with everything working. After the trial, or once a licence has passed its end date, graphics still go to air but carry a watermark: a “Cuelio Studio · Unlicensed” badge in the corner and “UNLICENSED” across the picture. The status bar shows the days left in the trial, or that the watermark is on; click it to open the licence settings.

1. Open **Settings** → **Licence**.
2. Paste the licence key from your purchase email (it starts with `CSRP-`) and press **Activate**. The computer needs the internet once, while it activates.
3. The panel then shows who the app is licensed to, the package and the date it is valid until.

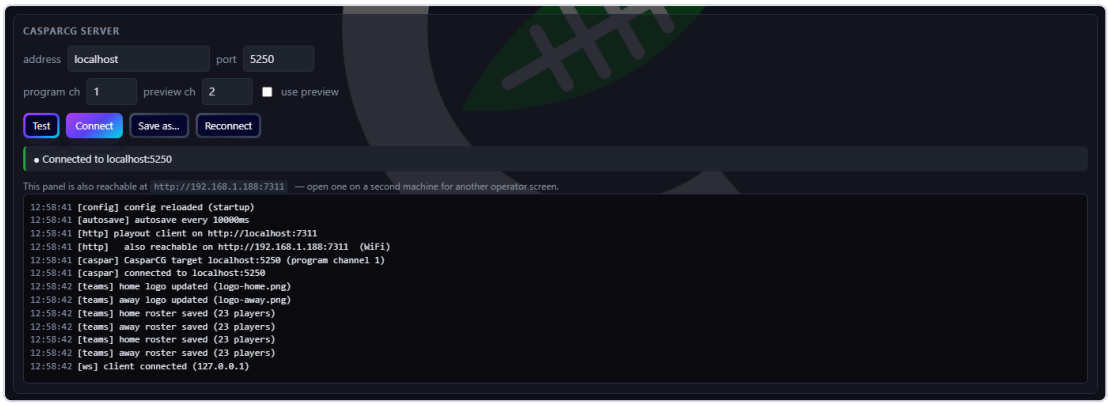
Each key works on **two computers**. To move it to a new computer, press **Deactivate** on the old one first. Licences run for 3 months, 6 months, 1, 2 or 3 years, or a lifetime.



Settings → Licence during the trial: this computer’s ID, the key box and Activate.

Connecting to CasparCG

1. Open **Settings** → **CasparCG server**.
2. Enter the server’s address and port (normally `5250`), and the program channel (normally `1`). Optionally tick **use preview** and give a preview channel.
3. Press **Connect**. The light in the bottom-right corner of the bar along the foot of the window turns green and shows the server, channel and server version.
4. Use **Save as…** to keep the server in the saved list for next time.



Settings → CasparCG server: address, port and channels, with the connection log below.



Connected: the green light, the server, and the channel and server version from CasparCG.

Installing the graphics

The graphics are HTML templates in the `RUGBY` folder. CasparCG 2.5 can load them straight from the panel over http, or you can copy the `RUGBY` folder (including `_shared`) into the CasparCG server's `template` folder. **Settings** → **Template health** lists any configuration problems, and **Template preview** opens every graphic with sample data in your browser, without sending anything to CasparCG.

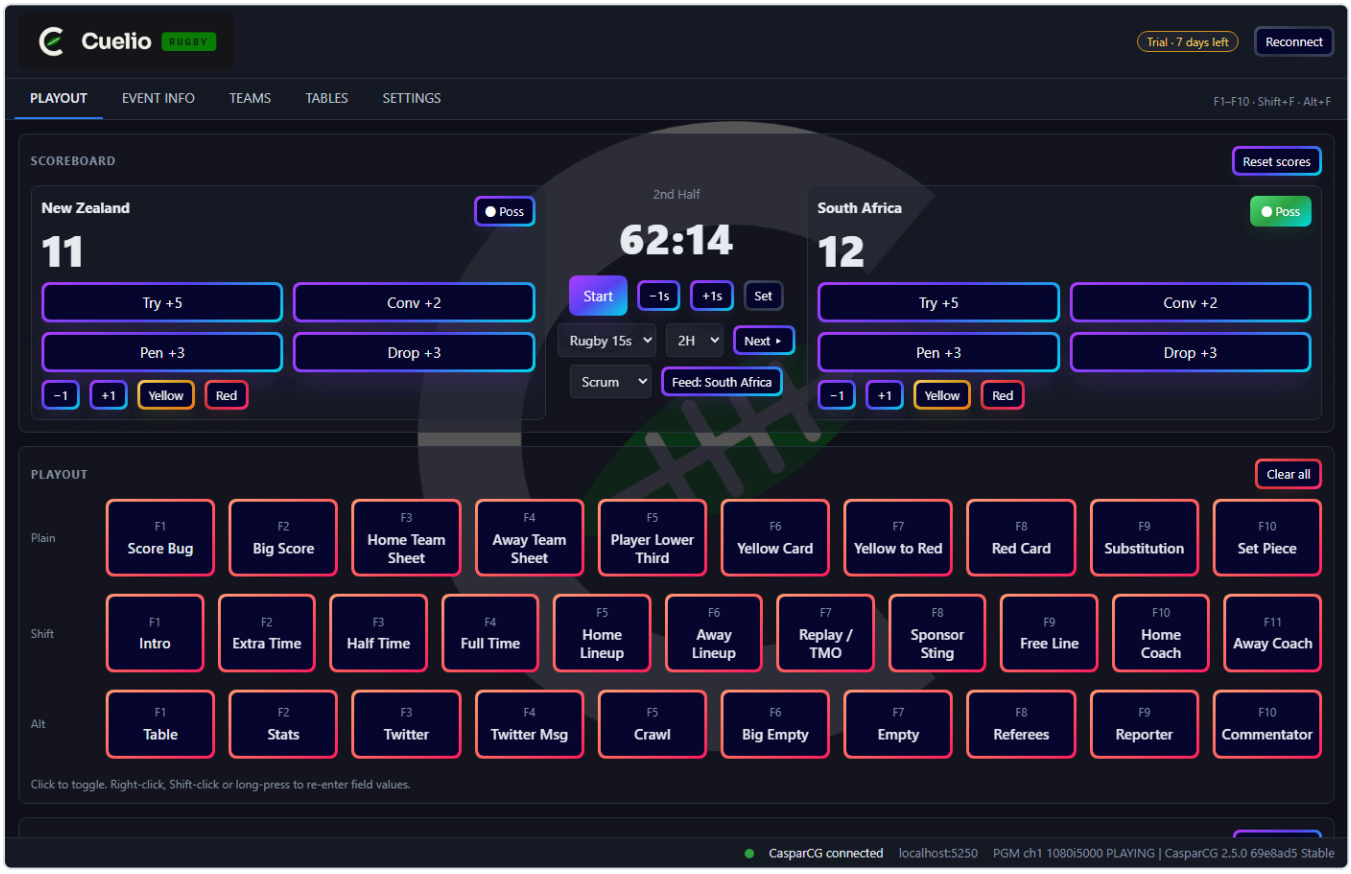


Settings → Template health, with the Template preview button.

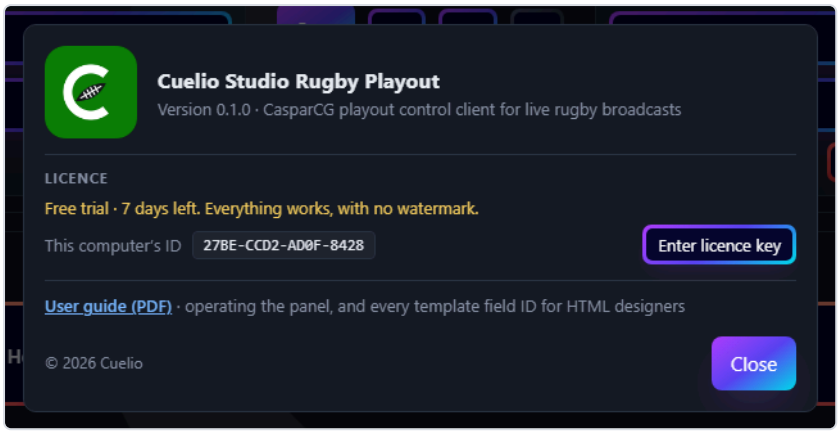
2. The panel at a glance

- **Status bar** (pinned along the top with the tabs): the Cuelio logo (click it for About and this guide), the trial or licence tag, and **Reconnect**.
- **Bottom bar** (pinned along the foot): the CasparCG connection light, the server, and the channel and server version, in the right-hand corner.
- **PLAYOUT**: the scoreboard, the match clock, the playout grid and the cards and events list.
- **Event Info**: teams and fixture (names, colours, logos, kits, finishes) and officials and on-air text.
- **Teams**: the two team sheets, the player picked for the Player lower third, and substitution pairs.
- **Tables**: the league table and the match stats.
- **Settings**: the licence, the CasparCG server, autosave and backups, template health and sponsors.

Everything saves as you leave a field, and the server is the single source of truth: a second screen sees every change at once.



The PLAYOUT tab during a match: scoreboard and clock above, the playout grid below.

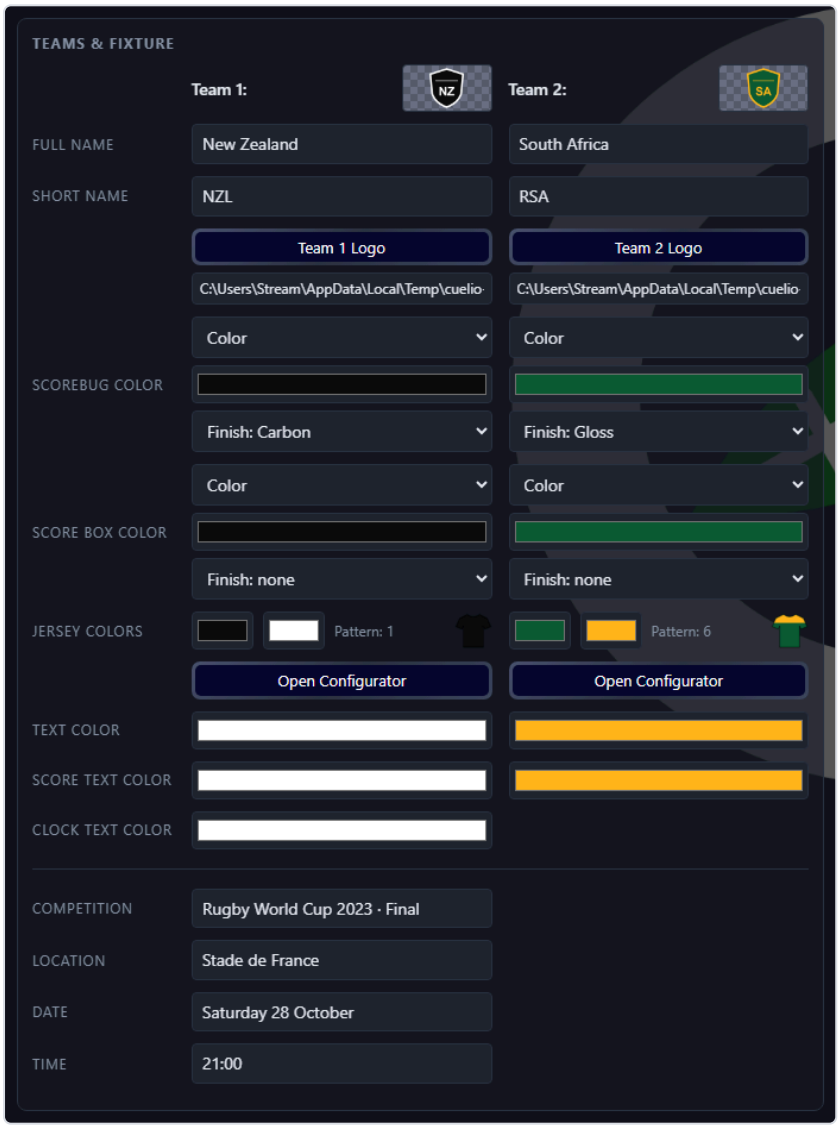


Click the Cuelio logo for About, with the link to this guide.

3. Before the match

Event Info → Teams & fixture

- **Full name** and **Short name** (three letters, for the score bug) for each side, and the team **logo** (upload, or a path the CasparCG server can see).
- **Scorebug color** and **Score box color**: a colour, an uploaded texture image, or none (the template's own colour). In colour mode a **Finish** can be laid over it: None, Gloss, Gradient, Carbon, Mesh, Stripes, Diagonal, Hoops, Pinstripe.
- **Jersey colors** and the **Configurator**: primary and secondary colour and the pattern (1 Plain, 2 Hoops, 3 Stripes, 4 Halves, 5 Sash, 6 Yoke).
- **Text color**, **Score text color** and the **Clock text color**.
- **Competition**, **Location**, **Date** and **Time**, used by the Intro and Big Empty.



Event Info → Teams & fixture: names, logos, score bug colours and finishes, kits and the fixture.

Event Info → Officials & on-air text

- The four **Referees** (Referee, two assistants, TMO) for the Referees graphic.

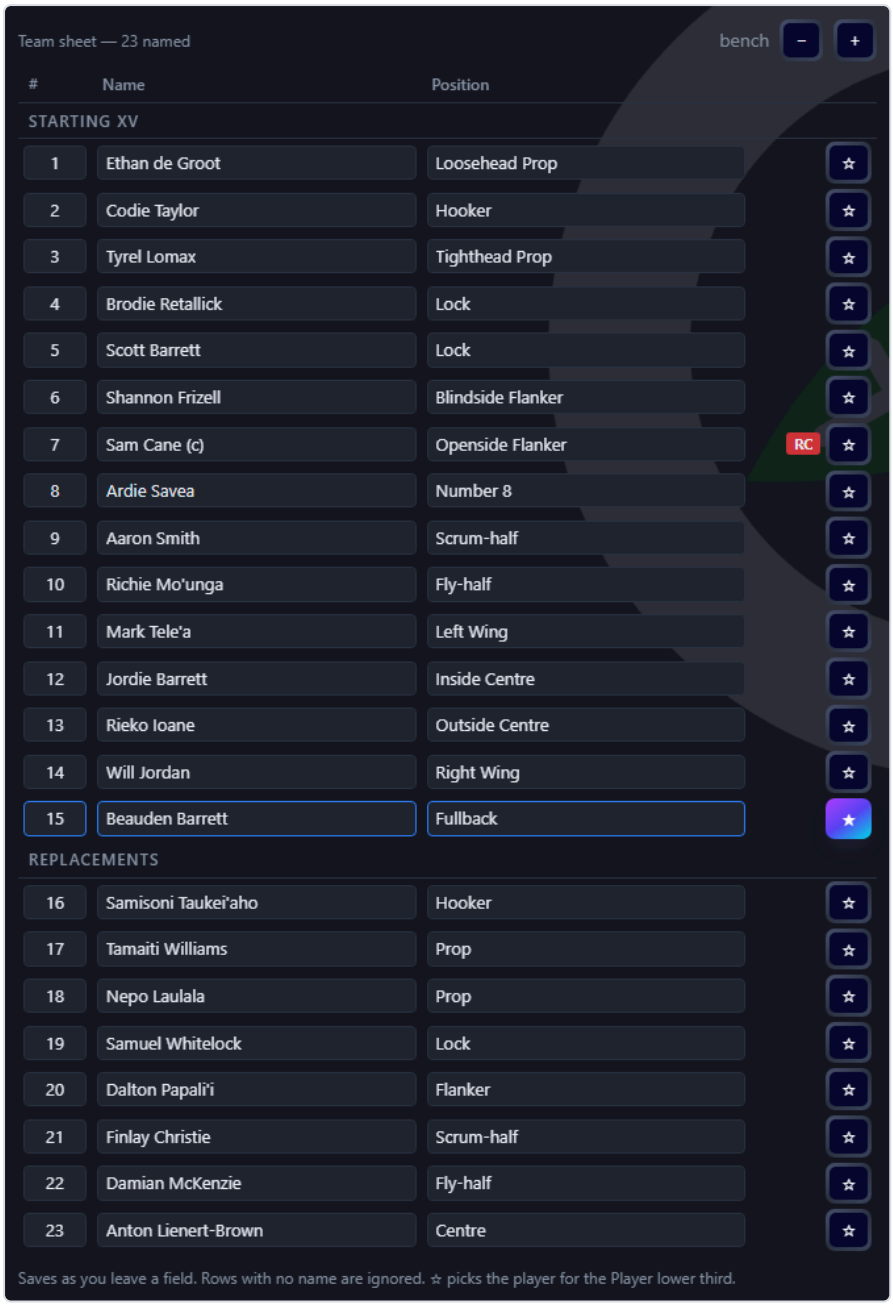
- **Commentator** names and title, **Reporter** name and title, and each side's **Coach** and coach title. These graphics take everything from here and go on air without a form.
- Two **Free lines**, three **Crawl** lines (the radio button picks the active one) and the **Big Empty** title and text.



Event Info → Officials & on-air text.

Teams

- Type each **team sheet**: Starting XV and replacements, with shirt number, name and position. Add "(c)" after the captain's name for a captain badge. Use **Import roster CSV** to load a sheet in one go.
- ☆ on a row picks that player for the **Player lower third**; their name, number, position, team name, logo and colour all follow.
- **Substitutions**: add numbered pairs (Player off from the starters, Player on from the replacements). The next pair to make is marked **Next**.



A team sheet. ★ marks the player picked for the Player lower third; RC shows a red card booked against No. 7.

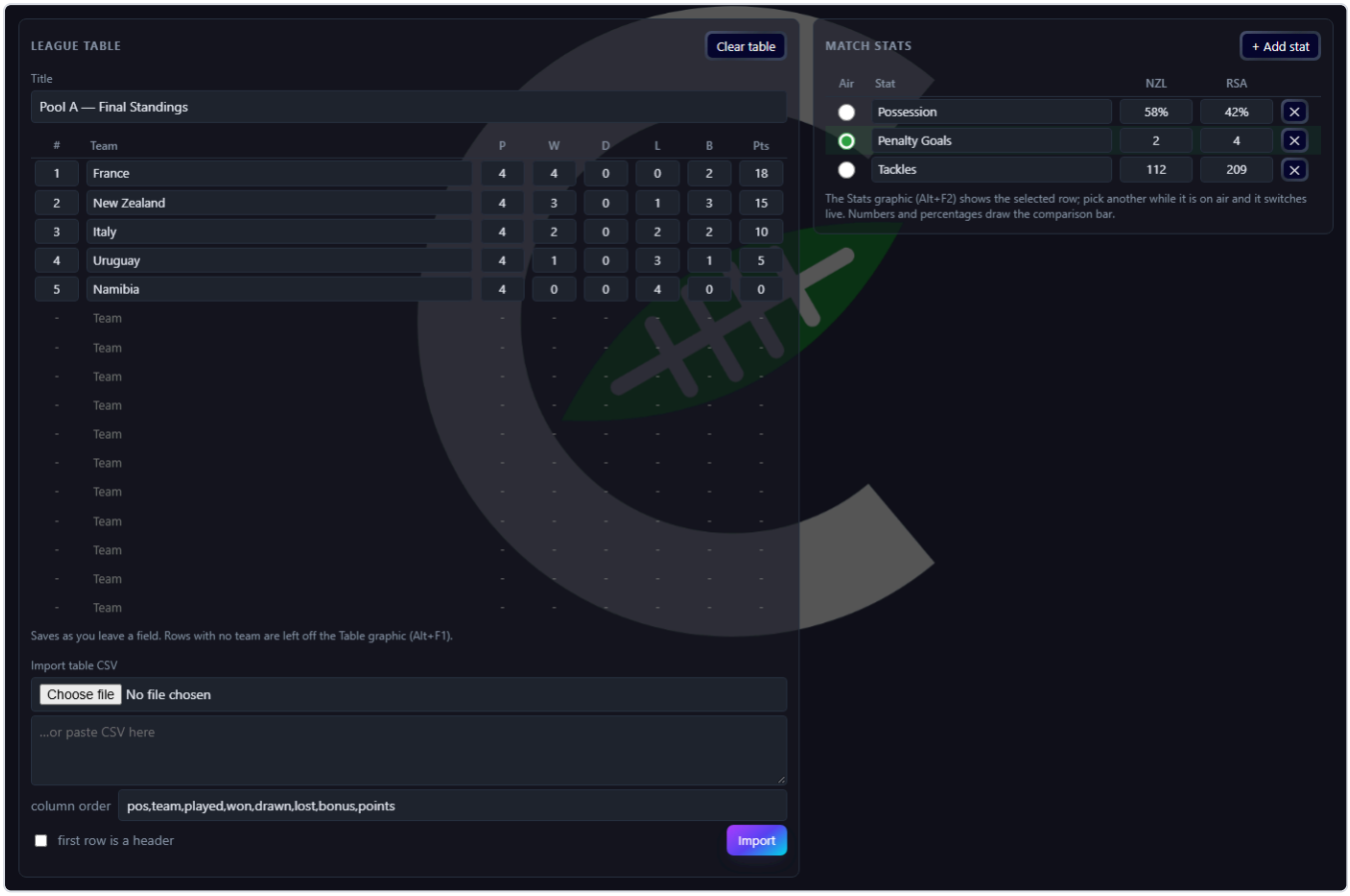


Substitution pairs: Player off from the starters, Player on from the replacements; the next to make is marked Next.

Tables

- **League table**: a title and up to 16 rows (position, team, played, won, drawn, lost, bonus, points), typed or imported from CSV.

- **Match stats**: stat rows with a home and an away value; the radio button picks the stat the Stats graphic shows.



The Tables tab: the league table on the left, match stats on the right.

4. During the match

Scoring and the clock

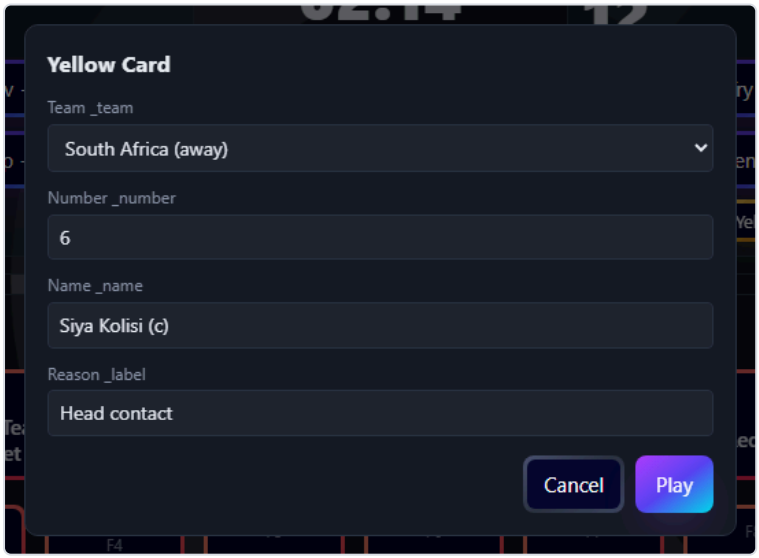
- **Try +5**, **Conv +2**, **Pen +3**, **Drop +3** and **−1 / +1** for each side. Graphics on air that show the score update straight away.
- **Start / Stop** runs the clock in match time (also the **Space** bar); **−1s / +1s** and **Set** correct it. Pick the format (15s or 7s) and the period, or press **Next ▸**.
- **Poss** marks the side in possession; the score bug lights a bar under that team. The set-piece type and the **Feed** side drive the Set Piece graphic.



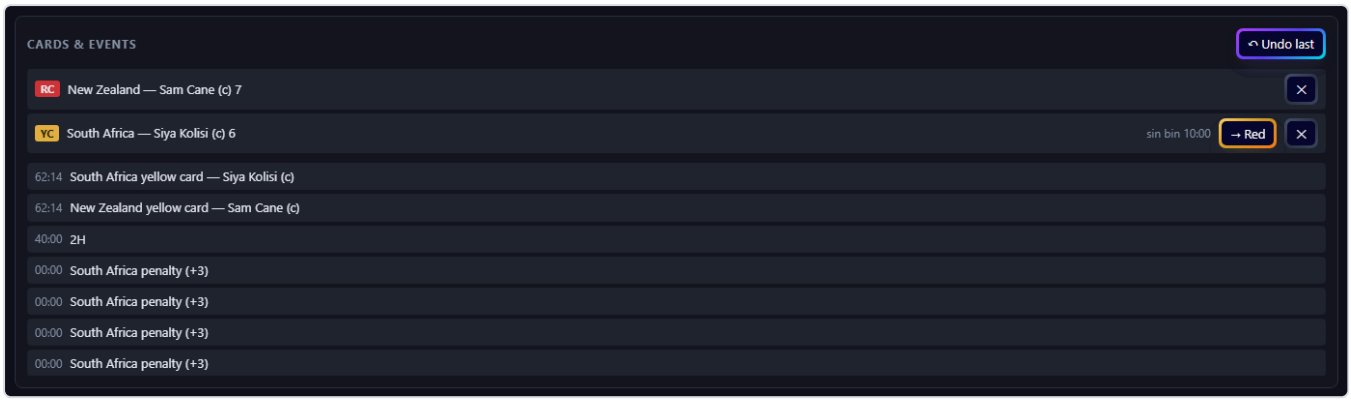
The scoreboard: score buttons for each side, the clock and period in the middle, possession and set piece.

Cards

The Yellow Card, Yellow to Red and Red Card buttons ask for the team, then a shirt number or a name: the other fills in from that team's sheet, and the image is the team's logo. Putting the graphic on air **books the card**: it appears in **Cards & events** with the sin-bin countdown, and as a YC / RC chip on the player's row on the Teams tab. Yellow to Red turns the player's yellow into a red. Re-airing a graphic never books the card twice.



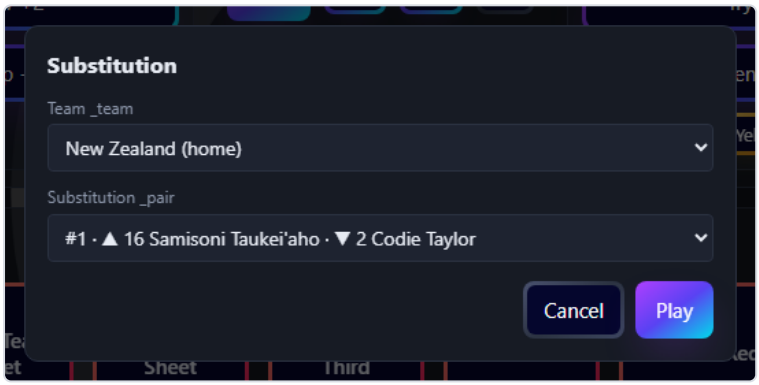
The Yellow Card form: South Africa picked and shirt 6 typed, so the name filled itself in from the team sheet.



Cards & events: each booked card with its sin-bin countdown, → Red to upgrade a yellow, ✕ to remove, and the event log.

Substitutions

Substitution asks for the team and the pair, starting on the next pair not yet made. Airing it marks the pair done and swaps the two players on the team sheet; ↶ beside a done pair undoes it.



The Substitution form, opened on the next pair not yet made.

Undo and reset

↶ **Undo last** under Cards & events reverses the last score or card. **Reset scores** zeroes both scores and leaves the clock and cards alone.

5. The playout grid and keyboard

Each button puts its graphic on air; pressing it again plays it out. A lit button is on air. Some graphics take others off when they go up, so two never sit on top of each other (for example Big Score and the score bug). A button with a form opens it the first time; **Shift-click** or **right-click** opens the form again to change what it shows.



The playout grid: three rows of buttons, fired by F-keys, Shift+F-keys and Alt+F-keys. A lit button is on air.

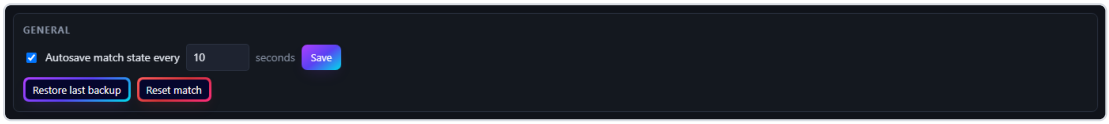
Plain	**F1** Score Bug	**F2** Big Score	**F3** Home Team Sheet	**F4** Away Team Sheet	**F5** Player Lower Third	**F6** Yellow Card	**F7** Yellow to Red	**F8** Red Card	**F9** Substitution	**F10** Set Piece	
Shift	**Shift+F1** Intro	**Shift+F2** Extra Time	**Shift+F3** Half Time	**Shift+F4** Full Time	**Shift+F5** Home Lineup	**Shift+F6** Away Lineup	**Shift+F7** Replay / TMO	**Shift+F8** Sponsor Sting	**Shift+F9** Free Line	**Shift+F10** Home Coach	**Shift+F11** Away Coach
Alt	**Alt+F1** Table	**Alt+F2** Stats	**Alt+F3** Twitter	**Alt+F4** Twitter Msg	**Alt+F5** Crawl	**Alt+F6** Big Empty	**Alt+F7** Empty	**Alt+F8** Referees	**Alt+F9** Reporter	**Alt+F10** Commentator	

Keyboard

- **F1–F11**, **Shift+F-key** and **Alt+F-key** fire the three rows of the grid.
- **Space** starts and stops the match clock (not while typing in a field).
- **Esc** closes a form or window.
- **Clear all** takes every graphic off air in one press.

6. Settings and safety nets

- **Autosave** keeps the match every few seconds. After an unexpected stop the app offers to restore it; **Restore last backup** does the same on demand.
- **Reset match** clears scores, clock, cards and the event log but keeps the teams.
- **Sponsors**: upload sponsor images into numbered slots for the Sponsor sting.
- If CasparCG drops and comes back, graphics that were on air are brought back up to date automatically.



Settings → General: autosave, restoring the last backup and resetting the match.

7. Chroma mode: playing out from one HDMI port

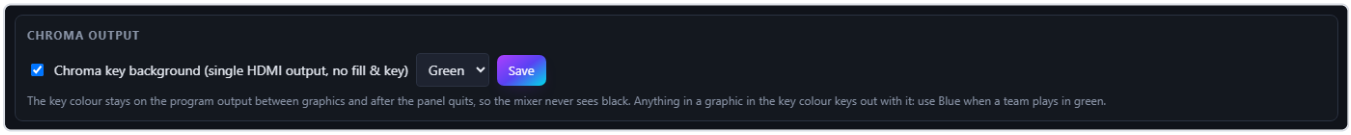
A broadcast playout normally sends the vision mixer two signals, **fill** (the graphics) and **key** (where they are see-through). A laptop usually has a single HDMI port and so no key signal. Chroma mode lets that one feed be keyed instead: the program output always sits on a solid key colour, and the mixer's chroma keyer removes it, leaving the graphics over the pictures.

What it does

- The key colour fills the program output **all the time**, not only while a graphic is on air. With nothing playing the output is solid green, never black, so the key never punches a black hole through the pictures.
- Every graphic plays over the colour as usual. Taking one off air, the Empty buttons and **Clear all** leave the colour in place.
- It stays up after the panel quits. Only switching chroma mode off removes it.
- If CasparCG restarts or the link drops, the colour goes back up first when it reconnects, then the graphics that were on air.
- The bottom bar shows **CHROMA · GREEN** (or **CHROMA · BLUE**) while it is on, so nobody mistakes the feed for fill & key.

Turning it on

1. Open **Settings** → **Chroma output**.
2. Tick **Chroma key background** and choose the key colour.
3. Press **Save**. The output changes straight away; so does switching colour or turning it off.



Settings → Chroma output: the switch and the key colour.

Choosing the key colour

Colour	Value	Use it when
Green (default)	`#00B140`	Most matches. Green keys the most cleanly on most mixers.
Blue	`#0047BB`	A team plays in green (South Africa, Ireland and others). Anything in a graphic that is the key colour is keyed out with the background, so a green team colour, logo or kit would go see-through.

Setting up CasparCG for the HDMI port

The panel decides *what* plays; CasparCG decides *where* the picture goes. In the server's `casparcg.config`, give the program channel a `<screen>` consumer that fills the HDMI display. With the laptop screen as display 1 and the HDMI output as display 2 (Windows **Settings** → **Display**, *Extend these displays*):

```
<channel>
  <video-mode>1080i5000</video-mode>
  <consumers>
    <screen>
      <device>2</device>            <!-- the HDMI display -->
      <windowed>false</windowed>
      <borderless>true</borderless>
      <always-on-top>true</always-on-top>
      <stretch>fill</stretch>
      <aspect-ratio>16:9</aspect-ratio>
    </screen>
    <screen>
      <device>1</device>            <!-- optional: a monitor window on the laptop -->
      <windowed>true</windowed>
    </screen>
  </consumers>
</channel>
```

Restart the CasparCG server after editing the file. Set the HDMI display to the same resolution as the channel (1920×1080), and turn off Windows notifications and the screen saver, which would otherwise appear on air.

Setting up the vision mixer

- Feed the laptop's HDMI into a mixer input and use it as a **chroma key** (not a luma or linear key) over the program pictures.
- Set the keyer to the same colour as the panel, then sample it from the output with nothing on air.
- Fire a graphic and adjust the keyer's clip and gain until the edges are clean. Soft edges and drop shadows blend toward the key colour; that is normal for chroma keying, and a little spill suppression on the mixer tidies it.

8. For template designers

Every graphic is an HTML page, 1920×1080 with a transparent background, in `templates/RUGBY/`. CasparCG loads it from the panel at `http://<panel>:7200/templates/RUGBY/<NAME>` (no `.html`), or from a copy of the folder on the server. The field IDs your page must read are listed for every graphic in chapter 9.

What the page must provide

- `update(data)`: receives the fields as a JSON string (CasparCG XML `<templateData>` is also accepted by the shared runtime). Called on the first ADD and again whenever a bound value changes while on air, with the full set of fields each time.
- `play()`: animate in. `stop()`: animate out. Nothing may stay visible after `stop()`: every visible element, backgrounds included, must animate out.
- `next()` and `remove()` should exist, even if they do nothing.

What the data looks like

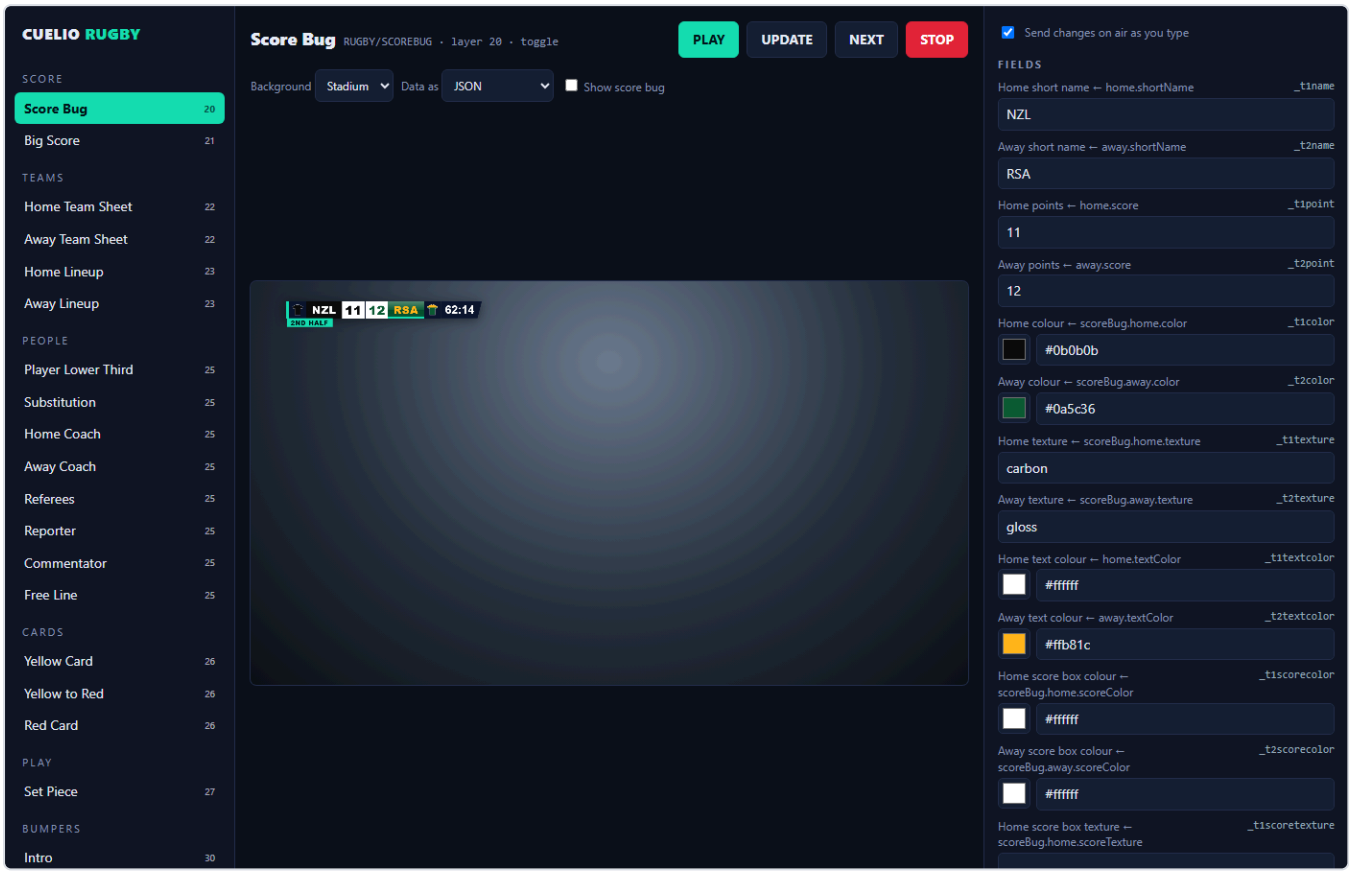
- Every value is a string. A blank string means “nothing to show”: hide the element rather than draw it empty.
- Images (logos, sponsors, uploads) arrive as `http://` URLs served by the panel.
- Colours are `#rrggbb`. Texture fields carry either an image URL or a finish name: `gloss`, `gradient`, `carbon`, `mesh`, `stripes`, `diagonal`, `hoops`, `pinstripe`.
- Jersey patterns arrive as Event Info’s numbers: `1` Plain, `2` Hoops, `3` Stripes, `4` Halves, `5` Sash, `6` Yoke.
- Score box modes are `color`, `texture` or `none`; possession is `home`, `away` or blank; team pickers send `home` or `away`.
- Fields whose source is a form are whatever the operator typed; automatic fields follow the match and update on air.

```
CG 1-25 ADD 1 "RUGBY/PLAYER" 1 "{\"_name\":\"Ross Byrne\",\"_number\":\"10\",\"_position\":\"Fly Half\",
  \"_label\":\"Leinster Rugby\",\"_image\":\"http://10.0.0.20:7200/logos/logo-
home.png\",\"_color\":\"#1b4ea1\"}"
```

The shared runtime (_shared/cuelio.js and cuelio.css)

- `data-f="_field"` fills an element’s text; `data-case="upper"` upper-cases it; `data-pop` pulses it when it changes.
- `data-img="_field"` sets an image; `data-empty="_a _b"` hides an element while all those fields are blank; `data-fit="min-px"` shrinks text to fit its box.
- `_t1color` / `_t2color` / `_color` and their text colours become the CSS variables `--t1`, `--t1-ink`, `--t2`, `--t2-ink`, `--c`, `--c-ink`.
- Elements with class `anim` plus a direction (`from-left`, `wipe`, `zoom` …) animate in on play and out on stop; `--d` and `--o` set the in and out delays in ms.
- Give `<body>` a `data-template="NAME"`. The whole graphic is drawn at full size and scaled by `--ui-scale` about its anchor `--ui-origin` in `cuelio.css`; keep anchors inside title safe (96 px in from the sides, 54 px from top and bottom).

> Test a template without CasparCG at `http://<panel>:7200/templates/preview/` : it lists every graphic, fills it with sample data, and has Play, Update and Stop buttons.



The template preview: pick a graphic on the left, edit its fields on the right, and Play, Update, Next or Stop it on the stage.

9. Template field reference

Generated from the app's template configuration (30 graphics on 25 template files). "Automatic" fields are filled by the app from the path shown and follow the match while on air; the other fields come from the button's form. Field IDs are case-sensitive.

Where automatic values come from

Path starts with	Meaning
`home.` / `away.`	The team on Event Info: name, short name, score, colours, logo, kit, coach.
`scoreBug.home.` / `scoreBug.away.`	The score bug look picked on Event Info (colour, texture or finish, score box mode), already resolved for the bug.
`event.`	Event Info: officials, reporter, commentators, free lines, crawl, Big Empty, clock text colour.
`clock.`, `halfLabel`, `possession`, `setPiece.`, `sinBinLabel`	The live match on the PLAYOUT tab.
`competition`, `venue`	Event Info: Competition and Location.
`selectedPlayer.`	The player picked with ☆ on the Teams tab, with their team's name, logo and colour.
`formTeam.`	The team picked in the button's form (Home or Away).
`formSub.`	The substitution pair picked in the button's form, looked up on that team's sheet.
`table.`, `stats.`	The Tables tab: the league table and the selected match stat.

Score

Score Bug `scoreBug`

Button: **Score Bug** · **F1**
Template: `RUGBY/SCOREBUG` · layer 20
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

Field ID	Label	Where the value comes from
`_t1name`	Home short name	Automatic, from `home.shortName`
`_t2name`	Away short name	Automatic, from `away.shortName`
`_t1point`	Home points	Automatic, from `home.score`
`_t2point`	Away points	Automatic, from `away.score`
`_t1color`	Home colour	Automatic, from `scoreBug.home.color`
`_t2color`	Away colour	Automatic, from `scoreBug.away.color`

Field ID	Label	Where the value comes from
`_t1texture`	Home texture	Automatic, from `scoreBug.home.texture`
`_t2texture`	Away texture	Automatic, from `scoreBug.away.texture`
`_t1textcolor`	Home text colour	Automatic, from `home.textColor`
`_t2textcolor`	Away text colour	Automatic, from `away.textColor`
`_t1scorecolor`	Home score box colour	Automatic, from `scoreBug.home.scoreColor`
`_t2scorecolor`	Away score box colour	Automatic, from `scoreBug.away.scoreColor`
`_t1scoretexture`	Home score box texture	Automatic, from `scoreBug.home.scoreTexture`
`_t2scoretexture`	Away score box texture	Automatic, from `scoreBug.away.scoreTexture`
`_t1scoremode`	Home score box mode	Automatic, from `scoreBug.home.scoreMode`
`_t2scoremode`	Away score box mode	Automatic, from `scoreBug.away.scoreMode`
`_t1scoretextcolor`	Home score text colour	Automatic, from `home.scoreTextColor`
`_t2scoretextcolor`	Away score text colour	Automatic, from `away.scoreTextColor`
`_t1jersey1`	Home jersey primary	Automatic, from `home.jerseyPrimary`
`_t1jersey2`	Home jersey secondary	Automatic, from `home.jerseySecondary`
`_t1jerseypattern`	Home jersey pattern	Automatic, from `home.jerseyPattern`
`_t2jersey1`	Away jersey primary	Automatic, from `away.jerseyPrimary`
`_t2jersey2`	Away jersey secondary	Automatic, from `away.jerseySecondary`
`_t2jerseypattern`	Away jersey pattern	Automatic, from `away.jerseyPattern`
`_time`	Clock	Automatic, from `clock.display`
`_clocktextcolor`	Clock text colour	Automatic, from `event.clockTextColor`
`_half`	Period	Automatic, from `halfLabel`
`_possession`	Possession	Automatic, from `possession`

Big Score `bigScore`

Button: **Big Score** · **F2**
Template: `RUGBY/BIGSCORE` · layer 21
Toggle: first press plays it in, second press plays it out
Takes off: Score Bug, Player Lower Third, Substitution, Home Coach, Away Coach, Referees, Reporter, Commentator, Free Line, Yellow Card, Yellow to Red, Red Card, Extra Time, Half Time, Full Time

Field ID	Label	Where the value comes from
`_t1name`	Home name	Automatic, from `home.name`
`_t2name`	Away name	Automatic, from `away.name`
`_t1point`	Home points	Automatic, from `home.score`
`_t2point`	Away points	Automatic, from `away.score`
`_half`	Period	Automatic, from `halfLabel`
`_t1color`	Home colour	Automatic, from `home.color`

Field ID	Label	Where the value comes from
`_t2color`	Away colour	Automatic, from `away.color`
`_t1textcolor`	Home text colour	Automatic, from `home.textColor`
`_t2textcolor`	Away text colour	Automatic, from `away.textColor`
`_t1logo`	Home logo	Automatic, from `home.logo`
`_t2logo`	Away logo	Automatic, from `away.logo`

Teams

Home Team Sheet `homeTeamSheet`

Button: **Home Team Sheet** · **F3**
Template: `RUGBY/TEAMSHEET` · layer 22
Toggle: first press plays it in, second press plays it out

Field ID	Label	Where the value comes from
`_teamname`	Team	Automatic, from `home.name`
`_teamlogo`	Logo	Automatic, from `home.logo`
`_color`	Team colour	Automatic, from `home.color`
`_textcolor`	Team text colour	Automatic, from `home.textColor`
Team sheet rows, n = 1 to 23, from the home team on the Teams tab: starters first, then replacements, each in shirt-number order. An empty slot is sent as blanks.		
`_p{n}number` `_p1number … _p23number`	Shirt number	Automatic, from the team sheet
`_p{n}name` `_p1name … _p23name`	Player name	Automatic, from the team sheet
`_p{n}position` `_p1position … _p23position`	Position	Automatic, from the team sheet (may be blank)
`_p{n}image` `_p1image … _p23image`	Player image	Automatic, from the team sheet (may be blank)

Away Team Sheet `awayTeamSheet`

Button: **Away Team Sheet** · **F4**
Template: `RUGBY/TEAMSHEET` · layer 22
Toggle: first press plays it in, second press plays it out

Field ID	Label	Where the value comes from
`_teamname`	Team	Automatic, from `away.name`
`_teamlogo`	Logo	Automatic, from `away.logo`
`_color`	Team colour	Automatic, from `away.color`
`_textcolor`	Team text colour	Automatic, from `away.textColor`
Team sheet rows, n = 1 to 23, from the away team on the Teams tab: starters first, then replacements, each in shirt-number order. An empty slot is sent as blanks.		
`_p{n}number` `_p1number … _p23number`	Shirt number	Automatic, from the team sheet
`_p{n}name` `_p1name … _p23name`	Player name	Automatic, from the team sheet
`_p{n}position` `_p1position … _p23position`	Position	Automatic, from the team sheet (may be blank)

Field ID	Label	Where the value comes from
`_p{n}image` `_p1image … _p23image`	Player image	Automatic, from the team sheet (may be blank)

Home Lineup `homeLineup`

Button: **Home Lineup** · **Shift+F5**
Template: `RUGBY/LINEUP` · layer 23
Toggle: first press plays it in, second press plays it out

Field ID	Label	Where the value comes from
`_teamname`	Team	Automatic, from `home.name`
`_teamlogo`	Logo	Automatic, from `home.logo`
`_color`	Team colour	Automatic, from `home.color`
`_textcolor`	Team text colour	Automatic, from `home.textColor`
Team sheet rows, n = 1 to 15, from the home team on the Teams tab: starters first, then replacements, each in shirt-number order. An empty slot is sent as blanks.		
`_p{n}number` `_p1number … _p15number`	Shirt number	Automatic, from the team sheet
`_p{n}name` `_p1name … _p15name`	Player name	Automatic, from the team sheet
`_p{n}position` `_p1position … _p15position`	Position	Automatic, from the team sheet (may be blank)
`_p{n}image` `_p1image … _p15image`	Player image	Automatic, from the team sheet (may be blank)

Away Lineup `awayLineup`

Button: **Away Lineup** · **Shift+F6**
Template: `RUGBY/LINEUP` · layer 23
Toggle: first press plays it in, second press plays it out

Field ID	Label	Where the value comes from
`_teamname`	Team	Automatic, from `away.name`
`_teamlogo`	Logo	Automatic, from `away.logo`
`_color`	Team colour	Automatic, from `away.color`
`_textcolor`	Team text colour	Automatic, from `away.textColor`
Team sheet rows, n = 1 to 15, from the away team on the Teams tab: starters first, then replacements, each in shirt-number order. An empty slot is sent as blanks.		
`_p{n}number` `_p1number … _p15number`	Shirt number	Automatic, from the team sheet
`_p{n}name` `_p1name … _p15name`	Player name	Automatic, from the team sheet
`_p{n}position` `_p1position … _p15position`	Position	Automatic, from the team sheet (may be blank)

Field ID	Label	Where the value comes from
`_p{n}image` `_p1image … _p15image`	Player image	Automatic, from the team sheet (may be blank)

People

Player Lower Third `player`

Button: **Player Lower Third** · **F5**
Template: `RUGBY/PLAYER` · layer 25
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

Name, number, position, team logo and stripe colour all follow the player picked on the Teams tab, on air too, so the button takes no form.

Field ID	Label	Where the value comes from
`_image`	Team logo	Automatic, from `selectedPlayer.logo`
`_color`	Team colour	Automatic, from `selectedPlayer.color`
`_name`	Name	Automatic, from `selectedPlayer.name`
`_number`	Number	Automatic, from `selectedPlayer.number`
`_position`	Position	Automatic, from `selectedPlayer.position`
`_label`	Strap text	Automatic, from `selectedPlayer.teamName`

Substitution `substitution`

Button: **Substitution** · **F9**
Template: `RUGBY/SUBSTITUTION` · layer 25
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

Asks for the team and one of the substitution pairs planned on the Teams tab, starting on the next one not yet made. Airing it marks the pair done and swaps the two players on the team sheet.

Field ID	Label	Where the value comes from
`_team`	Team	Home or Away, picked in the button's form; sent as "home" / "away"; default `home`
`_pair`	Substitution	A substitution pair number, picked in the button's form
`_inName`	In name	Automatic, from `formSub.inName`
`_inNumber`	In number	Automatic, from `formSub.inNumber`
`_outName`	Out name	Automatic, from `formSub.outName`
`_outNumber`	Out number	Automatic, from `formSub.outNumber`
`_color`	Team colour	Automatic, from `formTeam.color`

Home Coach `homeCoach`

Button: **Home Coach** · **Shift+F10**
Template: `RUGBY/COACH` · layer 25
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

Name, strap, colour and logo all come from the home team on Event Info and refresh on air as they are edited there, so the button takes no form.

Field ID	Label	Where the value comes from
`_name`	Name	Automatic, from `home.coach`
`_teamname`	Team	Automatic, from `home.name`
`_color`	Team colour	Automatic, from `home.color`
`_label`	Coach title	Automatic, from `home.coachTitle`
`_image`	Image	Automatic, from `home.logo`

Away Coach `awayCoach`

Button: **Away Coach** · **Shift+F11**
Template: `RUGBY/COACH` · layer 25
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

As Home Coach, for the away side.

Field ID	Label	Where the value comes from
`_name`	Name	Automatic, from `away.coach`
`_teamname`	Team	Automatic, from `away.name`
`_color`	Team colour	Automatic, from `away.color`
`_label`	Coach title	Automatic, from `away.coachTitle`
`_image`	Image	Automatic, from `away.logo`

Referees `referees`

Button: **Referees** · **Alt+F8**
Template: `RUGBY/REFEREES` · layer 25
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

All four officials from Event Info; a slot with no name is left off.

Field ID	Label	Where the value comes from
`_ref1role`	Referee role	Automatic, from `event.officials.0.role`
`_ref1name`	Referee	Automatic, from `event.officials.0.name`
`_ref2role`	AR 1 role	Automatic, from `event.officials.1.role`
`_ref2name`	Assistant referee 1	Automatic, from `event.officials.1.name`
`_ref3role`	AR 2 role	Automatic, from `event.officials.2.role`
`_ref3name`	Assistant referee 2	Automatic, from `event.officials.2.name`
`_ref4role`	TMO role	Automatic, from `event.officials.3.role`
`_ref4name`	TMO	Automatic, from `event.officials.3.name`

Reporter `reporter`

Button: **Reporter** · **Alt+F9**
Template: `RUGBY/REPORTER` · layer 25

Toggle: first press plays it in, second press plays it out
Takes off: Big Score

Name and title both come from Event Info and refresh on air as they are edited there, so the button takes no form.

Field ID	Label	Where the value comes from
`_name`	Name	Automatic, from `event.reporter`
`_label`	Title	Automatic, from `event.reporterTitle`

Commentator `commentator`

Button: **Commentator** · **Alt+F10**
Template: `RUGBY/REPORTER` · layer 25
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

Both commentators' names and the title come from Event Info and refresh on air as they are edited there, so the button takes no form.

Field ID	Label	Where the value comes from
`_name`	Names	Automatic, from `event.commentatorsText`
`_label`	Title	Automatic, from `event.commentatorTitle`

Free Line `freeLine`

Button: **Free Line** · **Shift+F9**
Template: `RUGBY/REPORTER` · layer 25
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

Rides the Reporter lower third: free line 1 as the name, free line 2 as the strap, both from Event Info.

Field ID	Label	Where the value comes from
`_name`	Line 1	Automatic, from `event.freeLines.0`
`_label`	Line 2	Automatic, from `event.freeLines.1`

Cards

Yellow Card `yellowCard`

Button: **Yellow Card** · **F6**
Template: `RUGBY/YELLOWCARD` · layer 26
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

Pick the team, then a shirt number or a name: the other fills in from that team's sheet. The image is the picked team's logo. Going to air books the yellow card against the player.

Field ID	Label	Where the value comes from
`_team`	Team	Home or Away, picked in the button's form; sent as "home" / "away"; default `home`
`_number`	Number	Typed in the button's form; fills from the `_team` team's sheet when the name is entered
`_name`	Name	Typed in the button's form; fills from the `_team` team's sheet when the shirt number is entered
`_image`	Team logo	Automatic, from `formTeam.logo`
`_label`	Reason	Typed in the button's form
`_sinbin`	Sin-bin length	Automatic, from `sinBinLabel`

Yellow to Red `yellowToRed`

Button: **Yellow to Red** · **F7**
Template: `RUGBY/YELLOWTORED` · layer 26
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

As Yellow Card: pick the team, then a number or a name. Going to air turns the player's yellow into a red, or books a red if they had none.

Field ID	Label	Where the value comes from
`_team`	Team	Home or Away, picked in the button's form; sent as "home" / "away"; default `home`
`_number`	Number	Typed in the button's form; fills from the `_team` team's sheet when the name is entered
`_name`	Name	Typed in the button's form; fills from the `_team` team's sheet when the shirt number is entered
`_image`	Team logo	Automatic, from `formTeam.logo`
`_label`	Reason	Typed in the button's form

Red Card `redCard`

Button: **Red Card** · **F8**
Template: `RUGBY/REDCARD` · layer 26
Toggle: first press plays it in, second press plays it out
Takes off: Big Score

Send-off shape, as Yellow Card: pick the team, then a number or a name, and going to air books the red card against the player. The Navis user guide documents Red Card with substitution-shaped fields (_inName/_inNumber/_outName/_outNumber), likely a copy-paste error in the source doc; that shape is kept in alternativeFields.

Field ID	Label	Where the value comes from
`_team`	Team	Home or Away, picked in the button's form; sent as "home" / "away"; default `home`
`_number`	Number	Typed in the button's form; fills from the `_team` team's sheet when the name is entered
`_name`	Name	Typed in the button's form; fills from the `_team` team's sheet when the shirt number is entered
`_image`	Team logo	Automatic, from `formTeam.logo`
`_label`	Reason	Typed in the button's form

Play

Set Piece `setPiece`

Button: **Set Piece** · **F10**
Template: `RUGBY/SETPIECE` · layer 27
Toggle: first press plays it in, second press plays it out

Scrum and lineout share one template via _type. Split into two templates/buttons if the real graphics are separate.

Field ID	Label	Where the value comes from
`_type`	Type	Automatic, from `setPiece.type`
`_team`	Feeding team	Automatic, from `setPiece.teamName`
`_color`	Feeding team colour	Automatic, from `setPiece.teamColor`
`_textcolor`	Feeding team text colour	Automatic, from `setPiece.teamTextColor`

Openers, bumpers and stings

Intro `intro`

Button: **Intro** · **Shift+F1**
Template: `RUGBY/INTRO` · layer 30
Momentary: first press plays it in, second press plays it out

Field ID	Label	Where the value comes from
`_t1name`	Home name	Automatic, from `home.name`
`_t2name`	Away name	Automatic, from `away.name`
`_t1logo`	Home logo	Automatic, from `home.logo`
`_t2logo`	Away logo	Automatic, from `away.logo`
`_competition`	Competition	Automatic, from `competition`
`_venue`	Venue	Automatic, from `venue`
`_date`	Date	Automatic, from `event.date`
`_kickoff`	Kick-off time	Automatic, from `event.time`
`_t1color`	Home colour	Automatic, from `home.color`
`_t2color`	Away colour	Automatic, from `away.color`
`_t1textcolor`	Home text colour	Automatic, from `home.textColor`
`_t2textcolor`	Away text colour	Automatic, from `away.textColor`

Extra Time `extraTimeBumper`

Button: **Extra Time** · **Shift+F2**
Template: `RUGBY/BUMPER_ET` · layer 30
Momentary: first press plays it in, second press plays it out
Takes off: Big Score

Field ID	Label	Where the value comes from
`_t1name`	Home name	Automatic, from `home.name`
`_t2name`	Away name	Automatic, from `away.name`
`_t1point`	Home points	Automatic, from `home.score`
`_t2point`	Away points	Automatic, from `away.score`
`_t1color`	Home colour	Automatic, from `home.color`
`_t2color`	Away colour	Automatic, from `away.color`
`_t1textcolor`	Home text colour	Automatic, from `home.textColor`
`_t2textcolor`	Away text colour	Automatic, from `away.textColor`

Half Time `halfTimeBumper`

Button: **Half Time** · **Shift+F3**
Template: `RUGBY/BUMPER_HT` · layer 30

Momentary: first press plays it in, second press plays it out
Takes off: Big Score

Field ID	Label	Where the value comes from
`_t1name`	Home name	Automatic, from `home.name`
`_t2name`	Away name	Automatic, from `away.name`
`_t1point`	Home points	Automatic, from `home.score`
`_t2point`	Away points	Automatic, from `away.score`
`_t1color`	Home colour	Automatic, from `home.color`
`_t2color`	Away colour	Automatic, from `away.color`
`_t1textcolor`	Home text colour	Automatic, from `home.textColor`
`_t2textcolor`	Away text colour	Automatic, from `away.textColor`

Full Time `fullTimeBumper`

Button: **Full Time** · **Shift+F4**
Template: `RUGBY/BUMPER_FT` · layer 30
Momentary: first press plays it in, second press plays it out
Takes off: Big Score

Field ID	Label	Where the value comes from
`_t1name`	Home name	Automatic, from `home.name`
`_t2name`	Away name	Automatic, from `away.name`
`_t1point`	Home points	Automatic, from `home.score`
`_t2point`	Away points	Automatic, from `away.score`
`_t1color`	Home colour	Automatic, from `home.color`
`_t2color`	Away colour	Automatic, from `away.color`
`_t1textcolor`	Home text colour	Automatic, from `home.textColor`
`_t2textcolor`	Away text colour	Automatic, from `away.textColor`

Replay / TMO `replayLabel`

Button: **Replay / TMO** · **Shift+F7**
Template: `RUGBY/REPLAY` · layer 31
Toggle: first press plays it in, second press plays it out

Field ID	Label	Where the value comes from
`_label`	Label	Picked from a list in the button's form; options: `REPLAY`, `TMO REVIEW`, `SLOW MOTION`

Sponsor Sting `sponsorSting`

Button: **Sponsor Sting** · **Shift+F8**
Template: `RUGBY/SPONSOR` · layer 32
Momentary: first press plays it in, second press plays it out

May need zero fields if the real template is a pure animated sting.

Field ID	Label	Where the value comes from
`_image`	Sponsor slot	A sponsor slot picked in the button's form; sent as the image URL

Data and text

Table `table`

Button: **Table** · **Alt+F1**
Template: `RUGBY/TABLE` · layer 24
Toggle: first press plays it in, second press plays it out
Takes off: Score Bug

Field ID	Label	Where the value comes from
`_title`	Title	Automatic, from `table.title`
Table rows, n = 1 to 16, from the Tables tab. Rows with no team are sent as blanks.		
`_r{n}pos` `_r1pos … _r16pos`	pos	Automatic, from the league table
`_r{n}team` `_r1team … _r16team`	team	Automatic, from the league table
`_r{n}played` `_r1played … _r16played`	played	Automatic, from the league table
`_r{n}won` `_r1won … _r16won`	won	Automatic, from the league table
`_r{n}drawn` `_r1drawn … _r16drawn`	drawn	Automatic, from the league table
`_r{n}lost` `_r1lost … _r16lost`	lost	Automatic, from the league table
`_r{n}bonus` `_r1bonus … _r16bonus`	bonus	Automatic, from the league table
`_r{n}points` `_r1points … _r16points`	points	Automatic, from the league table

Stats `stats`

Button: **Stats** · **Alt+F2**
Template: `RUGBY/STATS` · layer 24
Toggle: first press plays it in, second press plays it out

One stat at a time: the row selected on the Tables tab, which switches on air as the selection changes.

Field ID	Label	Where the value comes from
`_title`	Stat	Automatic, from `stats.current.label`
`_t1name`	Home name	Automatic, from `home.name`
`_t2name`	Away name	Automatic, from `away.name`
`_t1value`	Home value	Automatic, from `stats.current.home`
`_t2value`	Away value	Automatic, from `stats.current.away`
`_t1color`	Home colour	Automatic, from `home.color`
`_t2color`	Away colour	Automatic, from `away.color`

Field ID	Label	Where the value comes from
`_t1textcolor`	Home text colour	Automatic, from `home.textColor`
`_t2textcolor`	Away text colour	Automatic, from `away.textColor`

Twitter `twitter`

Button: **Twitter** · **Alt+F3**
Template: `RUGBY/TWITTER` · layer 28
Toggle: first press plays it in, second press plays it out

Field ID	Label	Where the value comes from
`_handle`	Handle	Typed in the button's form
`_name`	Display name	Typed in the button's form
`_message`	Message	Typed in the button's form (several lines)
`_image`	Avatar	Typed in the button's form

Twitter Msg `twitterMessage`

Button: **Twitter Msg** · **Alt+F4**
Template: `RUGBY/TWITTERMSG` · layer 28
Toggle: first press plays it in, second press plays it out

Field ID	Label	Where the value comes from
`_message`	Message	Typed in the button's form (several lines)

Crawl `crawl`

Button: **Crawl** · **Alt+F5**
Template: `RUGBY/CRAWL` · layer 29
Toggle: first press plays it in, second press plays it out

Field ID	Label	Where the value comes from
`_message`	Crawl text	Typed in the button's form (several lines); starts from `event.crawlText`

Big Empty `bigEmpty`

Button: **Big Empty** · **Alt+F6**
Template: `RUGBY/BIGEMPTY` · layer 24
Toggle: first press plays it in, second press plays it out

Full-frame text card. Title and text come from Event Info and refresh on air as they are edited there.

Field ID	Label	Where the value comes from
`_competition`	Competition	Automatic, from `competition`
`_title`	Title	Automatic, from `event.bigEmpty.title`
`_text`	Text	Automatic, from `event.bigEmpty.text`

10. Troubleshooting

Symptom	What to check
Status light is red	The CasparCG address and port in Settings, that the server is running, and any firewall between them. Press **Reconnect**.
A graphic comes up blank	Open it in the Template preview. Check that the template is installed and that its field IDs match chapter 9.
A logo or sponsor is missing	Upload it in the panel so it is served over http; a `C:\…` path on another computer cannot be loaded by a remote CasparCG.
A button stays lit after the graphic has gone	Press it once more: that plays it out and resets the button.
Graphics are cut off on a TV	Keep anchors inside title safe; see chapter 8.
"UNLICENSED" on air	The trial has ended or the licence has run out: activate a key in **Settings** → **Licence**. The watermark goes as soon as it is activated.
"Already active on 2 computers"	Deactivate the licence on one of them (**Settings** → **Licence** → **Deactivate**), or ask Cuelio to free the seat of a computer you no longer have.
Chroma mode: the output is black, not green	Chroma mode is off, or CasparCG is not connected; check the CHROMA tag in the bottom bar and the connection light. If the server was restarted the colour returns when it reconnects.
Chroma mode: parts of a graphic disappear	They are the key colour. Switch **Settings** → **Chroma output** to the other colour, and set the mixer to match.
Chroma mode: nothing on the HDMI screen	The `<screen>` consumer's `<device>` number in `casparcg.config` does not match the HDMI display, or the displays are mirrored rather than extended.
The score went wrong	↶ **Undo last**, or **−1 / +1**. A crash is covered by autosave: restore from Settings.

© 2026 Cuelio · Cuelio Studio Rugby Playout 0.1.0