



bitdispatcher manual

A free 911 computer-aided dispatch system in one program, set up entirely in the web interface: the dispatcher screen, the recorded map and aerial photos, road-routed recommendations, move-ups, the mobile terminal and reports built from the ledger.

55 chapters · 6 figures · 8 thousand words · BioDream LLC

Contents

1. **Part 1. Dispatching**

1. Getting started
2. The dispatcher screen
3. Screens on several monitors
4. Incidents and call entry screen
5. Fire houses screen
6. Police status and mark-outs
7. Unit status screen
8. Map screen
9. Incident and situation screen
10. 911 calls
11. Incidents
12. Unit recommendations
13. Advisory suggestions
14. Move-up and cover
15. Unit statuses
16. The dispatch map
17. Aerial photos
18. The command line
19. Keyboard

2. **Part 2. In the field**

20. Mobile data terminal
21. Incident reports (RMS)
22. NCIC inquiries

3. **Part 3. Supervising**

23. The supervisor board
24. Operations report
25. The advisory codebook
26. Ledger and audit

4. Part 4. What 911 centres asked for

27. Transcripts and what was heard
28. Callers in another language
29. Where the caller is: device location and floor
30. Many callers, one incident
31. Non-emergency calls
32. Vehicle crash data
33. Drones
34. Radio transcripts
35. Incident summaries
36. Critical calls and care
37. Breaks and workload
38. Training
39. Quality review
40. Working through an outage
41. Line attacks (TDoS)
42. Hooks and connections

5. Part 5. Setting up

43. Units
44. Stations
45. Call types and run cards
46. Recording the map
47. County GIS
48. Settings
49. Integrations
50. BitVoice
51. Users and positions
52. Several sites
53. Premises and hazards
54. Migration from an earlier CAD
55. This manual

Dispatching

The dispatcher's screen and everything done from it: calls, incidents, units, the map and the command line.

Getting started

bitdispatcher is a computer-aided dispatch (CAD) system for 911 centres: one program with its own web interface. It creates and tracks incidents, recommends units by drive time over the road network, shows units, calls and incidents on a dispatch map with aerial photos, gives field crews a mobile terminal, and records every action in a tamper-evident ledger that reports are built from.

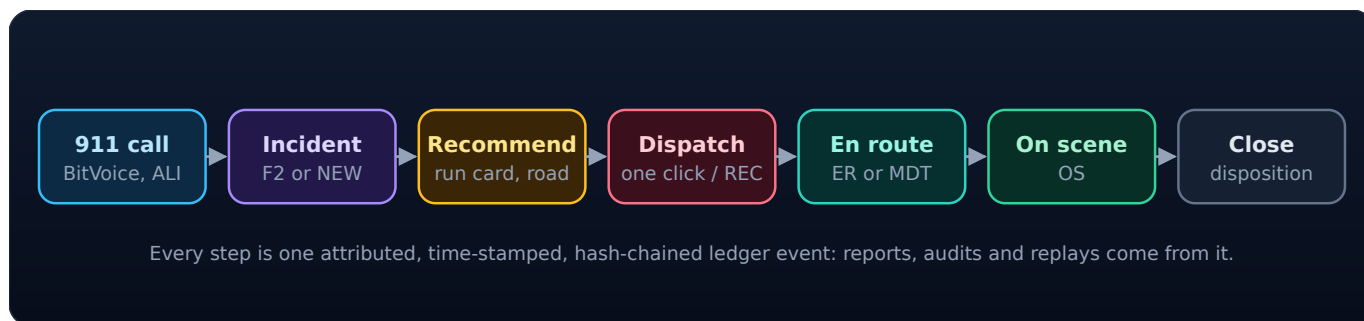


Figure 1. From a 911 call to a closed incident, and what each step writes to the ledger

The first hour

1. **Sign in** as the administrator with the password printed on first start. The position is the console you are at.
2. **Record the map:** Admin > Map, choose the centre and radius, press Record. About two minutes for 100 miles. [Open Map](#) (chapter 46)
3. **Stations and units:** where units are based and what they are. [Open Stations](#) (chapter 44) [Open Units](#) (chapter 43)

4. **Call types and run cards:** what each kind of incident needs. The defaults are a sensible start. [Open Call types](#) (chapter 45)
5. **Users:** dispatchers, supervisors and field crews. [Open Users](#) (chapter 51)
6. **Call handling:** connect BitVoice, or point any call-handling system's API spill and EIDO at bitdispatcher. [Open Integrations](#) (chapter 49)

bitdispatcher advises; the dispatcher decides. No feature dispatches a unit, closes an incident or changes a priority on its own.

Demo mode

Started with `-demo`, bitdispatcher seeds a fictional agency on the Roanoke, Virginia map (stations at real station positions, made-up units). With `bvsim`, a simulated BitVoice, calls ring and radios report positions; everything simulated is marked SIMULATED.

CHAPTER 2

The dispatcher screen

In the web interface: `/`

Everything a dispatcher needs is on one screen, and every part of it shows something useful.

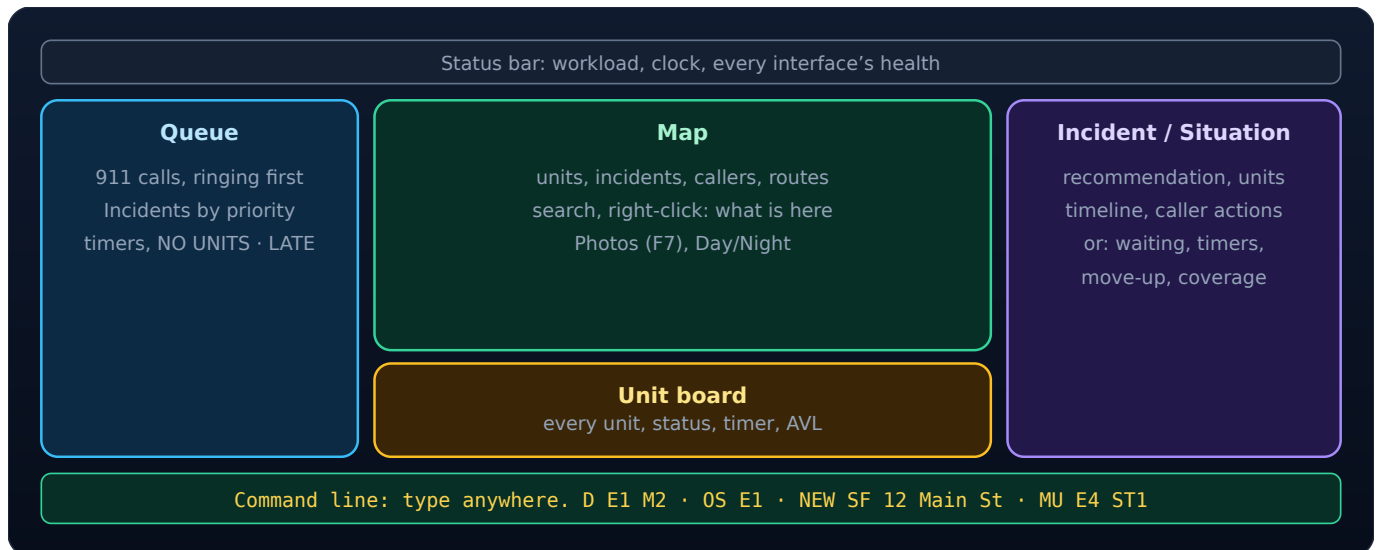


Figure 2. The dispatcher screen: calls and incidents on the left, the map in the middle, the selected incident on the right

Area	What it shows
Status bar	Calls waiting, pending and active incidents, units available by discipline, the clock, and the health of every interface (BitVoice, radio, ALI, map, ledger)
Queue (left)	911 calls, ringing first, with callback, location, uncertainty and age; incidents by priority with their timers
Map (centre)	Units by status, incidents, caller locations with uncertainty, road routes of recommended and assigned units with arrival times
Unit board (bottom)	Every unit by discipline: status, time in status, incident, AVL age; overdue timers in amber. Drag a unit onto an incident in the queue to send it; right-click a unit for its statuses, sending it to the selected incident, and moving it up
Right panel	The selected incident (recommendation, units, timeline, caller), or when nothing is selected the situation: what waits for units, overdue timers, move-ups (chapter 14), coverage by station, recent activity
Command line (bottom)	Type anywhere to start a command (see The command line (chapter 18))

Select an incident by clicking it, with `[` and `]`, or from the palette (`Ctrl K`). `Esc` clears the selection.

CHAPTER 3

Screens on several monitors

A position can spread its work over as many monitors as it has. Each monitor shows one **screen**: a window of the same live picture, arranged for one job. All of a position's windows share one sign-in and one selection: choose an incident on the queue screen and it opens on the incident screen and the map.

Open **Screens** with the monitor button on the status bar, from the command palette, or with **Ctrl+Shift+S**.

Screen	For
Incidents and call entry	911 calls and incidents beside the incident being worked; F2 for a new incident
Fire houses	Every station, its apparatus, who is covering and the empty houses
Police status and mark-outs	Every car: what, where, how long, and when its status check is due
Unit status	The unit board, large
Map	The map alone, following the selection
Incident and situation	The selected incident in full, or what needs attention
Supervisor board	Totals at a glance, read-only

- **Add a screen**, choose its **disciplines** (all, fire, EMS, police, other) and **Open** it. **Open all** opens every one.
- **Detect monitors** (Chrome and Edge) shows the monitors as the computer arranges them; give each screen its monitor and it opens there, filling it. In other browsers drag each window where you want it once: it opens there next time.
- The layout is kept on the workstation for the position, so the next shift at the desk gets the same screens. With **Offer to open these screens at sign-in**, the full console offers them in one click after signing in.
- The browser asks to allow pop-ups (and, in Chrome and Edge, window management) the first time.

Disciplines

A screen limited to some disciplines shows their units, and the incidents that need them: an incident's own call type, the units sent to it, and the units its run card asks for (a crash with injuries is on the fire screen and on the police screen). The status bar shows the screen's disciplines as a coloured chip. Every screen still shows every **emergency** and every 911 call not yet on an incident.

CHAPTER 4

Incidents and call entry screen

In the web interface: `/screen/calls`

The queue of 911 calls and incidents, with the incident being worked beside it at full height, the new-incident form (**F2**) and the command line. It is the call taker's and dispatcher's working screen when the map and units are on other monitors. Everything works as on the full console (see The dispatcher screen).

CHAPTER 5

Fire houses screen

In the web interface: `/screen/stations`

Every station as a card, with every unit that lives there or is covering there.

- **The bays:** one door per unit. Shut and green, it is in quarters and available; open, the stripe is the colour of its status.
- **The state:** IN QUARTERS (all home), PARTIAL, COVERED (a move-up is in) or EMPTY (no unit available in the house, with how long).
- **Tiles:** each unit's status, its time in it, and what it is doing (its incident, or where it is covering). Click, drag onto an incident or right-click as on the unit board.
- **Gaps:** a kind of unit the house should keep in quarters but does not, with the best **Move up** in one click (see Move-ups).
- Across the top: houses with a unit in, empty houses, units out, and each kind of unit available out of its total, with every status in a bar.

CHAPTER 6

Police status and mark-outs

In the web interface: `/screen/police`

Every police car in one table, with the map beside it: what it is doing, where, for how long, and when its **status check** is due.

- **Doing:** its incident (click to open it), or its **mark-out:** what it is doing on its own, such as a meal, court, fuel, a report or a follow-up.
- **Where:** the place given when it was marked out, its incident's address, or the nearest address to its AVL.
- **Status check:** a ring fills as the unit's time runs; amber near the end, red and flashing when the check is due, with how long past. A check is due after the status timer (Admin > Settings), or the mark-out's own time.

- **OK** records the status check (the unit is all right) and starts its timer again. **MO** marks it out; **AV** makes it available.
- The tiles across the top count available, on calls, marked out (by kind), out of service and checks due; click one to show only those.

From the command line

- **MO 2A12 MEAL 315 Church Ave** : mark out, with where. Without a place the unit's AVL position is kept. Add a note after a slash: **MO 2A15 COURT 315 Church Ave / Courtroom 2** .
MO alone lists the agency's kinds.
- **SC 2A12** : status check, all right. Several units at once: **SC 2A12 2A13** .

Mark-outs are also on every unit's right-click menu and in the unit details, and a crew marks itself out from the mobile terminal (**F7**). A mark-out that can be pulled (a meal, a report) is offered by the recommendation when no available unit can fill a line: it is never sent without a dispatcher choosing it. The kinds, their times and which can be pulled are set in Admin > Settings.

CHAPTER 7

Unit status screen

In the web interface: **/screen/units**

The unit board on its own, with larger tiles that also say what each unit is doing. Click, drag and right-click work as on the unit board of the full console.

CHAPTER 8

Map screen

In the web interface: **/screen/map**

The dispatch map alone. It follows the incident or call selected on any of the position's screens, and draws marked-out units where they were marked out when their AVL is not fresh.

CHAPTER 9

Incident and situation screen

In the web interface: **/screen/incident**

The selected incident at full height (its calls, units, timeline, premises, recommendation and report), or, with nothing selected, the situation: what is waiting, status timers past due, move-ups and activity.

CHAPTER 10

911 calls

Calls arrive from BitVoice's event stream, from EIDO pushed by any call-handling system, and from the ALI spill. The same call from several sources is merged into one line.

- **Ringling** calls show before they are answered, so the dispatcher sees what is coming.
 - **Location**: the caller's position and uncertainty circle, the address when there is one, and **STALE** when the location is older than the setting in Admin > Settings. Rebid asks for a new one (BitVoice).
 - **+ Incident** on a call creates an incident from it, with the location and callback filled in. Calls already attached show their incident.
 - Text to the caller, a video link and live video are on the call when BitVoice offers them.
 - Its [transcript](#) (chapter 27) and what was heard, [translation](#) (chapter 28), the [device location and floor](#) (chapter 29), [crash data](#) (chapter 32), callers [grouped](#) (chapter 30) and [non-emergency](#) (chapter 31) calls are described in their own chapters.
-

CHAPTER 11

Incidents

New incident: **F2**, the New button, + Incident on a call, right-click on the map (the address is filled in), a search result's New incident, or **NEW SF 2001 Crystal Spring Ave** on the command line. An address the recorded map knows exactly is placed on the map at once, so units are ranked by road straight away; anything else stays text to be placed.

The incident panel shows the recommendation, the units on it with their times, the caller and the timeline. Notes (**N text**) and priority notes (**PN text**) go on the timeline; priority notes are shown to every dispatcher and unit.

Close with a disposition (**CL Handled**); units still on it become available. Supervisors can reopen (**R0 #26-000123 reason**).

CHAPTER 12

Unit recommendations

For the selected incident bitdispatcher builds a recommendation from its call type's run card: for each line ("3 × engine", "1 × als"), the available units with that capability, closest first. Units already on the incident fill lines first, and scarce capabilities are filled before common ones so a multi-role unit is not spent where another would do.

- **Drive times** come from the recorded road network: posted or typical speeds, one-way streets obeyed, from the unit's fresh AVL position or its station. The reason says so: "1.7 mi by road, 4 min via Franklin Road".
- Outside the recorded map, or with no road between, the time is a straight-line estimate and is marked ~.
- A unit covering another station is ranked from where it is posted.
- Notes warn when no unit has a capability (consider mutual aid) or when sending the units would leave a station empty.

The recommended units are picked; **Dispatch** (`Ctrl Enter` , or `REC`) sends them. Click units to change the pick, **Use recommendation** to go back to it, or type `D E1 M2` . The ledger records what was recommended and what was sent.

CHAPTER 13

Advisory suggestions

Beside the incident and on the Situation panel, the advisory panel may suggest something: two incidents that may be the same, a call that may be about an open incident, words that do not fit the call type ("smoke" on a fire alarm, "not breathing" on a fall), a repeat address, a priority 1 still without a unit, a unit on scene not heard from.

- **Why**: open a suggestion to see the evidence it relied on (with its source: unit, caller, dispatcher, machine), what it assumed absent, and the rule in plain words.
- **Cannot determine** is an answer: when sources of equal weight disagree ("smoke" and "no smoke" from two callers), it says so and you decide.
- **Accept** runs the suggested command as you (change the type, attach the call, dispatch the recommendation), or just notes it. **Reject** with a one-click reason; it is not shown again. Both are recorded.
- It advises only. It never merges, retypes, reprioritises or dispatches by itself, and core dispatch works the same when it is off or down.

Move-up and cover

When a station has none of a kind of unit it normally houses available (its engine is out on a call), the Situation panel shows a **Move-up** box with the move that fills the gap: the nearest unit of that kind by road, from a station that can spare one.

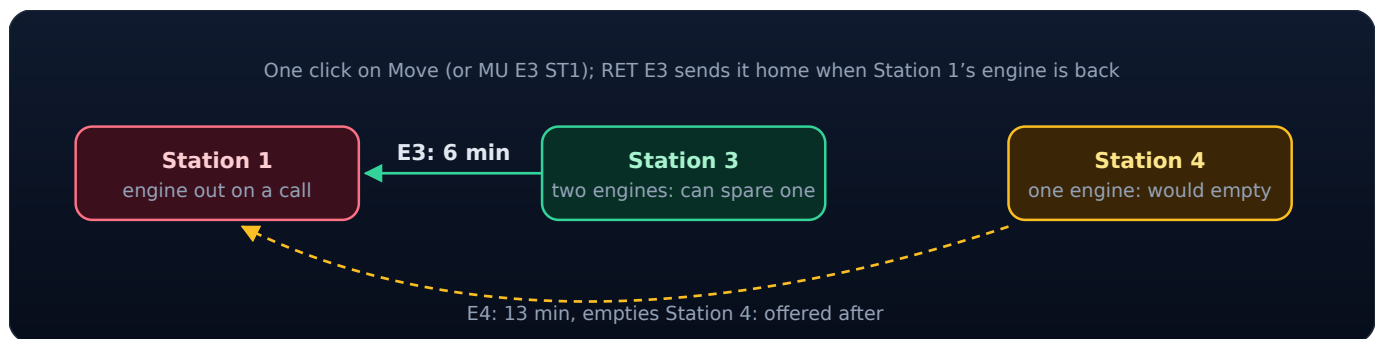


Figure 3. A move-up: Station 1 has no engine, so E4 covers it from Station 4

- **Move E3 from ST3** posts the unit at the empty station in one click (**MU E3 ST1**). It stays available and is recommended from there.
- A move that would leave its own station empty is offered only after moves that empty nothing, and says "empties ST4".
- When the station's own unit is back, a **Home** button sends the cover home (**RET E3**).
- Which kinds are kept covered is set per unit type in Admin > Settings (front-line engines, ladders, medics, ambulances and patrol by default).

Unit statuses

Code	Status	Meaning
AV	Available	In quarters or in service, can be sent
DP	Dispatched	Sent, not yet acknowledged
ER	En route	On the way
OS	On scene	Arrived
TR	Transporting	Taking a patient to hospital
AD	At destination	At the hospital
OOS	Out of service	Not available

Each status has a timer (Admin > Settings); a unit past its timer turns amber on the board and is listed under Status timers past due. Statuses are given on the command line (**ER E1** , **AV E1 M2**), from the unit's tile, or by the crew on the [mobile terminal](#) (chapter 20). AVL positions older than the stale setting are drawn dashed and not used to rank units.

CHAPTER 16

The dispatch map

The map is drawn from bitdispatcher's own [recorded map](#) (chapter 46): roads by class with route shields and one-way arrows, county and town lines, house numbers and hydrants at street level, fire, police, EMS and hospitals in their own colours.

- **Search** (top right): an address (**2001 Crystal Spring Ave**), an intersection (**Orange & Williamson** , **and** , **/** or **at**), a street, or a place. Arrow keys and Enter; New incident from a result. A house number not on file offers the nearest numbers on the same street.
- **Right-click** anywhere: what is here (the nearest address or road) and New incident here.
- **Routes**: for the selected incident, the road route of each recommended, chosen and assigned unit, with its arrival time on the unit.
- **Day / Night** switches the palette; **Photos** (**F7**) puts aerial photos under the streets; **Fit all** shows every unit and incident; OOS and Stations show or hide those.
- The dashed cyan line is the edge of the recorded map: beyond it the online map is shown, muted, and drive times are estimates.

Aerial photos

Dispatchers find places by rooftops ("the white house behind the church"). With **Photos** on (**F7** or the button on the map), aerial photos are the base and the streets, names and house numbers are drawn over them.

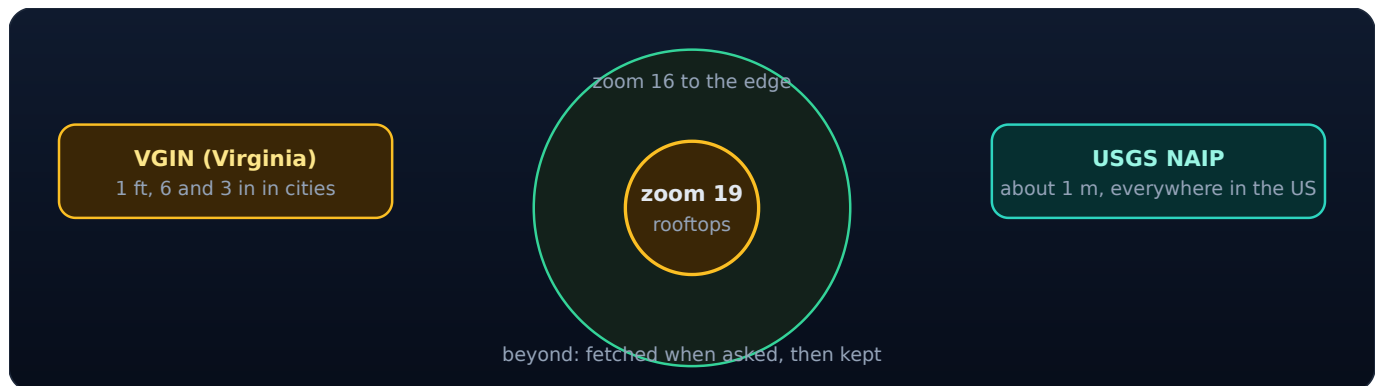


Figure 4. Aerial photos are recorded in two levels of detail

- Photos are recorded in Admin > Map: full detail near the centre, less detail out to the edge. Zooming in past what was recorded enlarges the photo below it, so the map never goes blank.
- The chip beside the button says when the photo at the map centre was taken, where the source publishes it ("2023, 12 inch ortho"): an old photo is never taken for today.
- Where nothing is recorded the server fetches the photo when asked and keeps it, unless that is switched off.

The command line

Type a letter anywhere on the dispatcher screen and the command line takes it (or press **/**). **Tab** completes, arrow keys recall earlier commands, results stay on screen. **HELP** lists every verb.

Command	Does
<code>D E1 M2</code>	Dispatch units to the selected incident (<code>D E1 #26-000123</code> names one)
<code>REC</code>	Dispatch the recommendation
<code>ER E1</code> · <code>OS E1</code> · <code>TR M1</code> · <code>AD M1</code>	En route, on scene, transporting, at destination
<code>AV E1 M2</code>	Available: clears them from their incident
<code>OOS E5 brakes</code>	Out of service with a reason
<code>N text</code> · <code>PN text</code>	Note, priority note
<code>NEW SF 12 Main St</code>	New incident (<code>NEW CHEST P1 ...</code> sets the priority)
<code>P 2</code> · <code>T VF</code>	Change priority or call type
<code>CL Handled</code> · <code>R0 #26-000123</code> <code>reason</code>	Close, reopen
<code>MU E4 ST1</code> · <code>RET E4</code>	Move up to cover a station; send home
<code>ATTACH <call> #26-000123</code>	Attach a 911 call to an incident
<code>ACK</code>	Acknowledge the oldest open emergency

CHAPTER 19

Keyboard

Key	Does
Any letter	Starts a command on the command line
/	Focus the command line
F2	New incident
Ctrl Enter	Dispatch the units picked for the selected incident
Ctrl K	Palette: go to an incident, unit, screen or chapter of this manual
[]	Previous and next open incident
Esc	Clear the selection
F7	Aerial photos on or off
F9	NCIC inquiry, with the selected incident as the reason
F1 – F6 (mobile terminal)	En route, on scene, transporting, at hospital, available, out of service

In the field

The mobile data terminal a crew uses from the vehicle.

CHAPTER 20

Mobile data terminal

In the web interface: `/mdt`

A crew signs in on the vehicle's laptop or tablet with a **field** account (Admin > Users) and the unit's call sign as the position (`E1`). The terminal opens at once.

- **The assignment:** type, priority, address in large type, the road route drawn on the map with its time and the roads it takes, the callback, the other units on it, and the notes from dispatch, newest first. Priority notes stand out.
- **Status buttons:** one tap each, with the usual next status highlighted; `F1` to `F6` on a keyboard.
- **Notes to dispatch** go on the incident's timeline as the unit's.
- **GPS:** the device's position goes to bitdispatcher as AVL every few seconds (allow location when the browser asks).
- **Emergency:** hold the button for two seconds. Dispatch sees the emergency banner and acknowledges it.
- **Photos** on the map, as on the console.

A crew acts only as its own unit. Dispatchers can open the terminal at `#/mdt` to see what a unit sees (the position names the unit).

CHAPTER 21

Incident reports (RMS)

In the web interface: `/rms`

Officers and dispatchers write incident reports at #/rms, or with **Report** on an incident (the incident's location and time are filled in). A crew on a mobile terminal writes its own from the car.

- **Sections:** when and where (location type, whether it was cleared), offences, people, property, vehicles and the narrative. Codes are shown in words and picked in one click; the screen asks only for what the offence needs (a weapon for an assault, the method of entry for a burglary, the drug for a seizure, the injury and the relationship to each suspect for a crime against a person).
- **Saved as you type:** a draft saves itself a moment after typing stops.
- **Before it is submitted** lists what is still missing for incident-based reporting: the FBI's NIBRS data elements and edits, with Virginia's additions (property quantity, the licence plate reader indicator, drug types). Submit is available once the list is empty.
- **Approval:** a supervisor other than the author approves it, or returns it with what to fix. Every save and decision is a ledger event.

The state file

Supervisors see **State IBR file** under the list. Choose the month and press **File**: bitdispatcher writes the month's approved reports in Virginia's incident-based reporting format (the FBI's position-based flat file with Virginia's additions, 2025.2 specification): Group A incidents with their offence, property, victim, offender and arrestee segments, Group B arrests, or a zero report when there is nothing. Upload it to the Virginia Crime Repository by the 15th of the next month. Reports that fail the checks are left out and listed. Making the file is a ledger event, with its SHA-256.

The settings need the agency's nine-character ORI (Admin > Settings). Before the first production upload, the Virginia State Police ask for a test file by email (dart@vsp.virginia.gov).

CHAPTER 22

NCIC inquiries

NCIC (on the reports screen, F9 on the dispatcher screen, or on the mobile terminal) sends an NCIC 2000 inquiry through the state's message switch: wanted person (**QW** : name with date of birth, or name, sex and race), vehicle (**QV** : plate and state, or VIN), gun (**QG**) and article (**QA**) by serial.

- Each inquiry needs a **purpose** code and a **reason** (the incident number): CJIS requires both.
- The ledger keeps who asked, when, the purpose, the reason and whether a record came back. The terms are kept only as a fingerprint and the answer not at all.
- **A hit is a lead**: confirm it with the entering agency before any action.
- Administrators set the agency's ORI and the switch in Admin > Settings: off, the **training simulator** (plates, names and serials starting with TEST return a record; every answer says SIMULATED), or a switch adapter by TCP.

Supervising

Reports built from the ledger, and the ledger itself.

The supervisor board

In the web interface: `/board`

The whole picture at a glance, for a supervisor's second monitor or a wall display. It only shows: nothing on it changes anything.

- **Across the top:** 911 calls not yet on an incident (red when one is ringing), incidents waiting for a unit with their priorities and the oldest wait, active incidents, units available out of those in service, units marked out by kind, status checks due, and empty stations.
- **Where the work is:** the map shaded by how much is open in each area (open incidents plus twice those waiting), with each area's open, waiting and available units. Areas are the county's emergency service zones or beats when they are loaded (Admin > County GIS); otherwise each station's area, the part of the map nearest it. An area with incidents waiting and no unit free, or two waiting, is **SWAMPED**.
- **By discipline:** a ring of every status with the number free, and each kind of unit available out of its total.
- **Busiest areas**, worst first, and **new incidents** over the last six hours by priority.
- **People:** each signed-in position's workload and breaks, and the care list after critical calls. See [Breaks and workload](#) (chapter 37) and [Critical calls and care](#) (chapter 36). A possible line attack shows across the top ([TDoS](#) (chapter 41)).

Choose the disciplines across the top, or open it from Screens with them. **Full screen** is beside the clock.

Operations report

In the web interface: </admin/reports>

Admin > Operations report builds the Daily Operations Report and shift debrief for a period from the ledger alone: nobody retypes what CAD already holds.

- **Periods:** this shift, last shift (from the shift change times in Settings), today, yesterday, last 7 days, or any range.
- **Contents:** 911 calls, incidents, dispatches and closes; incidents by hour, type, priority and disposition; response times (call to dispatch, dispatch to en route, call to on scene, on scene to clear) as median and 90th percentile; priority 1 incidents; incidents open at the end (the turnover); status timers run out, once per episode; emergency activations and how fast they were acknowledged; each person's actions; unit demand by phase and utilisation; system and configuration changes.
- **Drill-down:** click any number to see the ledger events it is made of.
- **Certify:** once the period has ended, a supervisor certifies it with optional context. The certification is a ledger event with the report's hash; if anything changes later, the report says it changed since it was certified.

Activity counts intentional actions only and is never a score: it shows workload, not performance.

CHAPTER 25

The advisory codebook

In the web interface: </admin/advisory>

Admin > Advisory panel holds the codebook: the rules the panel uses, in plain words, with their thresholds.

- **Mode:** off; **shadow** (evaluated and kept in the ledger, shown to supervisors only, so the rules can be judged before anyone relies on them); **live** (dispatchers see suggestions). A new system starts in shadow.
- **Source precedence:** drag to order unit, caller, dispatcher and machine evidence, strongest first. A stronger source outweighs a weaker one; equal sources that disagree give "cannot determine". A silent alarm panel never outweighs a person who can see smoke.
- **Rules:** switch each on or off, set its severity, distances, times, words and the call types they already fit, and edit its explanation.

- **Test** replays the recorded ledger for the last day, week, month or quarter with the rules on screen, saved or not, and counts what each would have suggested.
- **Save version** keeps the change with a note; every version is listed and can be restored in one click. Decisions (accepted, rejected and why) are listed: rejecting a suggestion is never treated as an error.

Agency scripts

JavaScript script and **Lua script** add a rule that is the agency's own code, versioned, tested and switched between shadow and live with the rest of the codebook. A script defines one function for its hook:

Hook	Function	Returns
Each open incident	<code>advise(i)</code>	nothing, or <code>{title, detail, severity, action: {label, command}}</code> (or a list): a suggestion; accepting runs the command on the incident, as the person who accepts
Unit recommendation	<code>recommend(r)</code>	<code>{prefer: [unit ids], notes: [text]}</code> : a preferred unit among a line's next best takes a place; notes are shown with the recommendation
Incident reports	<code>check(r)</code>	a list of <code>{field, message}</code> : shown as Agency checks beside the state's; they advise and do not stop a submission

An incident script sees the number, type and its name, priority, status, address and position, minutes since it was created, notes, units on it, hazards at its premises, caller name and number of calls. **Try it** runs the script once on the newest real incident (or report) and shows what it returned, what it logged with `log()`, and how long it took; **show the input** shows exactly what it saw.

- A script has no files, network, process, environment or code loading: only its input and `log()`.
- It always has a time limit (50 ms unless set, at most 1 s). It runs after the CAD's state is released, so a slow script never holds dispatch up.
- A script that fails or runs out of time is shown under its rule and in the test; the panel and the recommendation carry on without it and say so.

Ledger and audit

In the web interface: `/admin/ledger`

Every action that changes state, takes attention or creates accountability is an event in an append-only ledger: who (user, role, position, session), when, what, on which incident, unit or call. Each event carries the hash of the one before it, so a change anywhere breaks the chain.

- **Verify** checks the whole chain and reports the first broken link, if any.
- **Search** by type, person or text; events are newest first.
- The current state is rebuilt from the ledger at every start, and reports are built from it.

What 911 centres asked for

Transcripts and translation, callers grouped and screened, crash data, drones, care for the people, training, quality review, outages and line attacks.

Transcripts and what was heard

When call handling transcribes calls, each call shows its **transcript** live, the caller in white and the call-taker in blue, and **Heard on the call**: the details pulled out of it, so the call-taker types less.

- **What is pulled out**: the address, weapons, vehicles, plates, the direction someone went, how many people, injuries (not breathing, unconscious, bleeding), hazards (fire, trapped, gas, a dog, an electric car) and the caller's name. Urgent ones are red and come first, and show on the call in the queue.
- **Use**: on an address, places the call's incident there (a map search, exact when the map knows the number; check it on the map). Before there is an incident it copies the address.
- **Where transcripts come from**: BitVoice's live transcript when it sends one, its transcript of a recorded call afterwards, or any speech-to-text engine posting to the transcript hook ([Hooks](#) (chapter 42)).
- Rules, not a model, pull out the details: the same words always give the same details, and each shows the words it came from.

A transcript is what a machine heard. The call-taker's own judgement and questions still decide.

Callers in another language

A caller speaking Spanish is translated line by line on the transcript, marked **Spanish · translated**; **Show original** shows what was said.

- **The phrasebook** built into bitdispatcher translates the words of emergencies between Spanish and English, offline. When it knows too few of a line's words it says **gist only**: get an interpreter.
- **Say it in Spanish**: press a ready phrase (Help is on the way, Is the person breathing?, Get out of the house now...) or type in English; the translation shows as you type. It is kept on the transcript, texted when **also text it** is ticked (BitVoice), and the speaker button says it aloud on this console.
- **A language model** (optional, Admin > Integrations) translates what the phrasebook cannot, on an OpenAI-compatible chat API: one on your own server, or a service under your agency's agreement. What callers say is sent to it, so it is off until an administrator switches it on.

CHAPTER 29

Where the caller is: device location and floor

Each call says how its position was found: **device location** (the phone's own fix, often a few metres, the FCC's dispatchable location), GPS, cell sector or on file.

- **Floor**: a civic floor from call handling, and the **altitude** of a 3-D fix with its uncertainty, which tells which floor of a tall building.
- **Trail**: a caller who moves leaves a dotted trail on the map of every position received, the selected call's brighter.
- **Rebid** asks for a new position; **STALE** marks an old one (Admin > Settings).

CHAPTER 30

Many callers, one incident

When several people call about the same thing, the queue groups them under **N callers about one place**: calls near each other (300 m by default) within a short time (10 minutes).

- Make the incident from one call; the group then offers **Attach N to** that incident, one press for the rest.

- With an incident selected, the group offers to attach all of them to it.
 - Calls already on different incidents are never grouped together. Change the distance and time in [Admin > Settings](#) (chapter 48).
-

CHAPTER 31

Non-emergency calls

Calls on a **non-emergency line**, or whose caller says something that is seldom an emergency (a noise complaint, parking, a power outage), go to the **Non-emergency** list under the 911 calls, with the reason.

- Anything urgent said on the call (a weapon, not breathing, fire) keeps it an emergency, whatever else was said; so does crash data.
 - **It is an emergency** moves it back; a person's decision always stands and is recorded.
 - **Send the usual reply** texts the reply set in Settings, such as where to report online.
 - Set the lines, the words and the reply in [Admin > Settings](#) (chapter 48).
-

CHAPTER 32

Vehicle crash data

A vehicle's automatic crash notification arrives as a call with **CRASH DATA**: the vehicle, delta-V (the change of speed in the crash), the direction of the impact, rollover, airbags, occupants and belts, and its power (electric, hybrid).

- **Injury risk** is high for delta-V of 40 km/h (25 mph) or more, a rollover, or 20 km/h with no belt, several impacts or a side impact; medium for 20 km/h or any airbag. Suggestions follow: ALS, extrication, air medical, more than one ambulance, and for an electric car a word to the fire crews.
 - It is a rule of thumb from field-triage criteria to help pick the response, not a diagnosis. Change the thresholds in [Admin > Settings](#) (chapter 48).
 - Crash notifications reach bitdispatcher through the crash hook ([Hooks](#) (chapter 42)).
-

CHAPTER 33

Drones

With a drone system connected ([Admin > Integrations](#) (chapter 49)), an incident on the map shows **Send a drone**. bitdispatcher sends the drone system the incident, its type, priority, position and summary; the drone system decides which drone flies.

- **Launch rule:** incidents of call types marked **Send a drone** (Call types) ask by themselves, once each.
- The drone reports back to the drone hook: its status, position, battery and live picture. It flies on the map, its status goes on the incident's timeline, and **Live picture** opens its video.

CHAPTER 34

Radio transcripts

When BitVoice or the radio system transcribes transmissions, a unit's words go on its incident's timeline in italics (*E1: on scene, smoke showing*), so the record holds what was said on the air. The words come with BitVoice's radio events, or through the radio-words hook ([Hooks](#) (chapter 42)).

CHAPTER 35

Incident summaries

Every open incident has a **Summary**: what and where, what the callers said that matters (the urgent first), the priority notes, the hazards on file at the place, who is on scene and who is responding, and a drone over it.

- It is built from the record by rules, so it says nothing the record does not; it changes as the incident does.
- It shows on the incident panel (Copy puts it on the clipboard) and at the top of the crews' [mobile terminal](#) (chapter 20).

CHAPTER 36

Critical calls and care

Some calls stay with the people who take them. An incident is a **critical call** when its call type is marked critical (cardiac arrest, shooting and stabbing to begin with) or when words such as not breathing, CPR, officer down or infant are said on it.

- Everyone who worked it (the call-taker and the dispatchers who entered, sent or noted) goes on the **care list** on the [supervisor board](#) (chapter 23), with the reason.
- The supervisor **offers a break** to each person, **arranges a debrief** (with a note of when and where) and marks it **Done**. Each is recorded.
- Once the incident closes, each person sees a quiet line on their own console: that was a hard call, take a moment, and the support line set in Settings. **Take a break** puts them on a break.
- Set the words, the support line and the call types in [Admin > Settings](#) (chapter 48) and [Call types](#) (chapter 45).

CHAPTER 37

Breaks and workload

The cup on the status bar puts you **on a break** and back; the supervisor board shows it.

The board's **People** panel lists every signed-in position: calls in the last hour, actions, time since the last break, critical calls in four hours, and a state:

- **watch**: two hours without a break (Settings), a critical call, or a very busy hour;
- **relieve**: three hours without a break, or two critical calls.

It names the least busy positions free to relieve someone.

CHAPTER 38

Training

Tick **Training** when signing in. The console opens a board of its own, marked TRAINING in violet: the agency's units, stations, call types, premises and map, on a ledger of its own. Each trainee has their own board, even on a shared account; one left idle for two hours is cleared. Nothing done there reaches the live board, and administration is closed.

- **Scenarios** play calls, transcripts, crash data and radio traffic into it on the agency's own map: a house fire with three callers, a cardiac arrest in Spanish, an interstate crash with crash data, an armed robbery, and a flood of calls on the line.
- **Score** checks what was done: every caller on an incident, the call type, the units sent, a reply in the caller's language, and each incident's [quality review](#) (chapter 39).
- **Reset** empties your board and starts it again; other trainees' boards are untouched.

CHAPTER 39

Quality review

In the web interface: </admin/quality>

Every incident checked against the centre's standards:

Check	Standard (Settings)
Answered in time	15 s, NENA's call-answering standard
Entered in time	answer to incident, 90 s
Dispatched in time	entry to the first unit, 60 s, priority 1 and 2
Location verified, callback, disposition	captured
Questions asked	heard on the transcript: the address, the callback, and by discipline (breathing and awake for EMS; anyone inside for fire; weapons, description and direction for police)

Each incident gets a score over the checks that applied; the page lists them for a period, each call-taker's average and each check's pass rate. Press an incident for its checks. A review is a coaching aid: a missed question may simply not be on the transcript.

CHAPTER 40

Working through an outage

Each console keeps the last board it had. When the server cannot be reached, an amber bar says so and the console shows that board, as it was, with the time. A console reloaded during the outage still opens.

- **Offline log:** new incidents, notes and unit statuses are kept in this browser, in order.
- When the server is back the bar offers **Send them now:** each goes in marked "entered offline at 14:02 by jlee"; anything the server refuses stays, with the reason.
- For a server that never goes away, run bitdispatcher at two sites ([Several sites](#) (chapter 52)): each keeps working when the other is cut off.

CHAPTER 41

Line attacks (TDoS)

A telephony denial-of-service fills the lines so real callers cannot get through. bitdispatcher watches the calls and raises **Possible telephony denial-of-service** on every console and the supervisor board when:

- far more calls arrive in a minute than usual (five times the last hour's rate, at least 12, or the number set in Settings);
- one number calls five times or more in ten minutes (shown as ×5 on its calls);
- eight or more silent or very short calls arrive in five minutes.

A supervisor **acknowledges** it; it clears after ten quiet minutes. Keep answering: a real emergency can be among them.

CHAPTER 42

Hooks and connections

Other systems send to bitdispatcher with `Authorization: Bearer` and the hook token set in [Admin > Integrations](#) (chapter 49). Without a token the hooks are off.

Hook	Sends
<code>/api/hooks/transcript</code>	<code>{"call": "id", "speaker": "caller", "text": "...", "lang": "es"}</code> , or <code>lines</code> : a call's words
<code>/api/hooks/acn</code>	a crash notification: JSON or key=value, common telematics names (delta_v_mph, pdof, rollover, airbag_deployed...)
<code>/api/hooks/drone</code>	<code>{"id", "status", "lat", "lon", "battery", "incident", "videoUrl"}</code>
<code>/api/hooks/radio-text</code>	<code>{"unit" or "radioId", "channel", "text"}</code>

The drone system's launch address and token, the launch rule, and the optional translation model are set on the same screen.

Setting up

Everything is set up on these screens; there is no configuration file to edit.

CHAPTER 43

Units

In the web interface: `/admin/units`

Each unit has a call sign (`E1` , `M12` , `2A31`), a type (which gives its capabilities), a station, and optionally extra capabilities, the radio's unit ID and alias (so radio emergencies and positions find it) and an agency for mutual aid. New units start out of service.

CHAPTER 44

Stations

In the web interface: `/admin/stations`

Stations are where units are based. Their positions place units with no AVL, are where move-ups post covering units, and are where drive times start from when a unit's AVL is stale. Type the address and pick it from the recorded map's suggestions: the position fills in. Or type the position.

CHAPTER 45

Call types and run cards

In the web interface: `/admin/calltypes`

A call type is a kind of incident (`SF` structure fire, `CHEST` chest pain) with a default priority, a discipline and a run card: how many units of each capability it needs ("3 × engine, 1 × ladder, 1

× als"). Recommendations fill the run card. Capabilities come from unit types (Admin > Settings) and units.

Critical call puts the people who work one on the supervisor's [care list](#) (chapter 36); **Send a drone** has the drone launch rule ask for a [drone](#) (chapter 33).

CHAPTER 46

Recording the map

In the web interface: `/admin/map`

bitdispatcher records the OpenStreetMap data for a circle around your centre into its own store, then draws the dispatch map, routes units by road and finds addresses from that store with no internet.

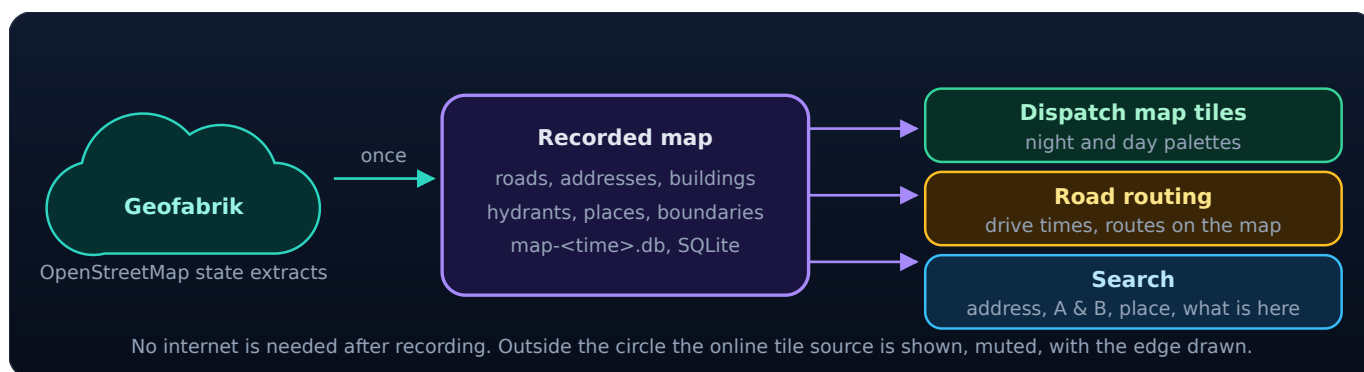


Figure 5. How the recorded map is made and used

1. Choose the centre (a preset, the console map centre, or a position) and the radius (100 miles covers a county and its mutual-aid neighbours).
2. The downloads the area needs are listed at once (state extracts from Geofabrik). Press **Record**.
3. Recording takes a few minutes; dispatching carries on with the current map, which stays until the new one is complete. Each recording is a ledger event.

Bring up to date applies OpenStreetMap's daily change files (Geofabrik publishes one a day per extract) since the map was recorded: a few megabytes instead of the whole extracts, then the store is built again. Record again for a new area, or when the map is more than a year behind; downloads already made are checked against their published checksums and reused.

Aerial photos

Below the map, **Aerial photos** records photos around the same centre: full detail within the radius you choose (zoom 19: rooftops and cars) and less out to the edge (zoom 16). The estimate (tiles, size, time) follows the form; tiles already recorded are skipped, so a stopped recording carries on where it was. The sources and their terms are listed: VGIN's Virginia Base Mapping Program for Virginia, USGS The National Map (NAIP, public domain) elsewhere.

CHAPTER 47

County GIS

In the web interface: `/admin/gis`

The county's own GIS is more exact than OpenStreetMap where the county keeps it. Admin › County GIS imports its layers for the recorded map's area into a store of their own, kept when the map is recorded again.

Kind	Used for
Address points	Searched first: an exact county address point wins over OpenStreetMap's; house numbers on the map
Hydrants	Drawn with their IDs at street level
Emergency service zones, beats	Drawn and labelled on the map
Common place names	Searched
Road centrelines	Their names are searched; with address ranges (from/to, left/right) they place a number that has no point of its own, on its side of the road, and give a point on a road its number. Shown as county range: estimate
Overture addresses	Open address points from the Overture Maps Foundation (in the US largely the National Address Database), for any area, without a county service. Used after the county's own, marked Overture

1. Paste the layer's ArcGIS REST address (`.../FeatureServer/0`), choose a preset (Virginia's statewide address points and road centrelines from VGIN, or Overture's open addresses),

- or upload a GeoJSON file or a CSV with latitude and longitude columns. The layer is read at once: its name, how many records lie in the area, and a guess at the kind and the fields.
2. Check the field mapping against the sample values shown beside each field, and press **Add layer**.
 3. **Import now** reads every layer that is on. The store in use stays until the new one is complete; each import is a ledger event.

Search results from county data are marked as such. ArcGIS layers are read with https only, in batches, so services that cap a query still give everything.

CHAPTER 48

Settings

In the web interface: `/admin/settings`

- **Agency** name and incident number prefix.
- **Status timers**: minutes in each status before the dispatcher is prompted for a status check.
- **Mark-outs**: the kinds a unit can be marked out for (code, name, status-check minutes, which units, whether it can be pulled for a call). See Police status and mark-outs.
- **Stale** ages for AVL and caller locations.
- **Shift changes** (`07:00 19:00`) for the operations report's shift periods.
- **Travel estimates** outside the recorded map: road factor and average speed.
- **Calls**: how near and how soon callers are [grouped](#) (chapter 30); the [non-emergency](#) (chapter 31) lines, words and reply.
- **People**: the [critical-call](#) (chapter 36) words, the support line, and when to suggest a [break](#) (chapter 37).
- **Quality review** standards and the questions to listen for ([Quality review](#) (chapter 39)); **crash data** thresholds ([Vehicle crash data](#) (chapter 32)); the [line-attack](#) (chapter 41) warning level.
- **Map beyond the recorded area**: the online tile source (fetched and cached by the server, never by browsers), and the map's starting centre and zoom.
- **Unit types**: code, name, discipline, capabilities, and whether stations keep them covered (move-up suggestions).
- **Dispositions** an incident can be closed with.

CHAPTER 49

Integrations

In the web interface: `/admin/integrations`

How bitdispatcher connects to call handling and radio. Each integration shows its live state and a checklist of the hooks in use.

- **BitVoice**: its address and API token, and its certificate pinned with Read it. See [BitVoice](#) (chapter 50).
- **EIDO**: the token a call-handling system uses to push EIDO to `/api/ng911/eido`.
- **Radio**: the secret for radio webhooks to `/api/radio/events`.
- **ALI spill**: the TCP address bitdispatcher listens on, and Test.
- **Hooks, drones and translation**: the hook token for transcripts, crash data, drone reports and radio words, the drone system, and the optional language model. See [Hooks and connections](#) (chapter 42).

CHAPTER 50

BitVoice

bitdispatcher uses every hook BitVoice offers a CAD, and needs none of them: without BitVoice, ALI and EIDO from any call-handling system work the same way.

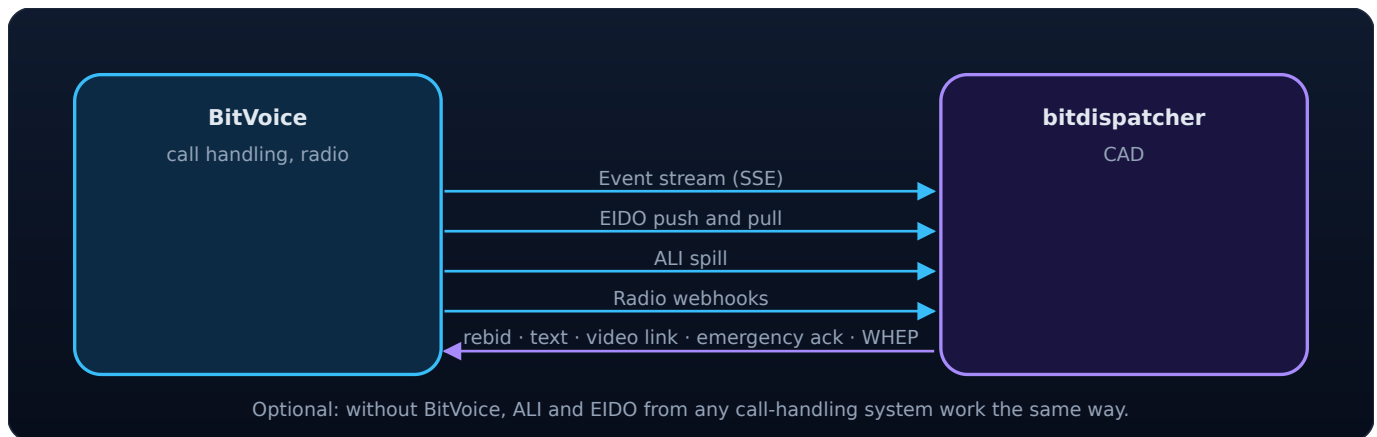


Figure 6. Every hook BitVoice offers a CAD, and which way each runs

Hook	Way	Used for
Event stream	BitVoice → bitdispatcher	Calls ringing, answered, ended; locations; resync after a gap
EIDO	both	Incident data pushed on answer and media changes; pulled when missed
ALI spill	BitVoice → bitdispatcher	The classic location record, merged with the same call
Radio webhooks	BitVoice → bitdispatcher	Emergency button, acknowledgement, transmissions, GPS
Callbacks	bitdispatcher → BitVoice	Rebid, text to caller, video link, live video, emergency acknowledgement

Delivery failures BitVoice reports show on the Situation panel.

CHAPTER 51

Users and positions

In the web interface: `/admin/users`

Role	Can
Viewer	See everything, change nothing
Field crew	The mobile terminal (chapter 20) for its own unit
Dispatcher	Work incidents, units and calls
Supervisor	Also reopen incidents, read the ledger and reports, certify reports
Administrator	Also change configuration and users

Passwords are generated or at least 12 characters. Five wrong passwords lock an account for 15 minutes. The positions list shows who is signed in where.

A public demonstration account. Tick **Public demonstration account** on a viewer, dispatcher or supervisor and the sign-in page offers **Sign in as the demo user**: one press signs a visitor in, with no password, at a position of their own (WEB- and four letters) unless they typed one. One account at most, never an administrator or a field crew, and wrong passwords

typed for it do not lock the button out. Use it only on a demonstration server: anyone who can reach the server can then work as that user.

CHAPTER 52

Several sites

In the web interface: `/admin/sites`

bitdispatcher runs at two or more sites, all live at once: a main centre and a backup centre, or neighbouring counties sharing a system. There is no primary to fail over from.

- **Each site writes its own ledger** and copies the others' over https, every second, each event checked against its site's hash chain before it is kept.
- **The same picture everywhere:** every site applies all events in the same order (by time, then site), so sites that have the same events show the same state, whatever order they arrived in.
- **Cut off, a site keeps working** on its own; when the link returns the events are exchanged and both sites settle on the same state. If two sites changed the same unit meanwhile, the later change wins, at both.
- **Incident numbers** carry each site's prefix (N26-000001, S26-000001), so two sites never give the same number.
- **Setting up:** give this site a name and prefix and save; copy its token and certificate fingerprint to the other sites. Add each other site with its address, its token and its fingerprint (**Read it** reads the certificate; check it against the one shown on that site). The status bar shows how many sites are linked.

A site's name and prefix are written into every event it makes, so they cannot change once in use.

CHAPTER 53

Premises and hazards

In the web interface: `/admin/premises`

A premise is what the centre knows about a place before anyone calls: **hazards** to the crews (a dog, oxygen in use, an armed occupant, a Knox box), **notes** and **contacts**. It is shown with every incident within 40 m of it, on the dispatcher's incident panel and on the mobile terminal, hazards first in red. An expired premise (a temporary hazard past its date) is shown as expired.

- **Add a premise:** type the address and pick it from the suggestions (that places it), click the common hazards or type others, add notes and contacts.
- **Search** by address, name, hazard or id; click one to change or remove it.
- Supervisors and administrators change premises; everyone sees them. Each change is a ledger event.

Premises from an earlier system come in through **Migration**.

CHAPTER 54

Migration from an earlier CAD

In the web interface: `/admin/migrate`

Admin > Migration brings an earlier CAD system's data over from the spreadsheets it exports, CSV or tab-separated (UTF-8 or Windows encoding).

1. **Drop the file** on the page. Its columns are matched to ours from their names, and every row is checked at once.
2. **Choose what it holds** if the guess is wrong: premises and hazards, incident history, or call types and run cards. Change any column in the table; the first row's value is shown beside it.
3. **Import** the rows that passed. The rows that did not are listed with why (no position and an address the map does not know, a date it cannot read).

Kind	Needs	Notes
Premises and hazards	an address (or latitude and longitude)	hazards and contacts separated by ; — an address without a position is found on the map (county points, then address ranges, then the map's)
Incident history	number and date	counts as calls before at an address: the incident panel shows the last five within 40 m over ten years, and the advisory panel's repeat-address rule counts them
Call types and run cards	code and run card	"2 engine; ladder:1; als": capabilities and counts. A capability no unit type has is pointed out

Importing the same file again replaces what it brought (premises by their id, call types by their code); history is added again, so import it once. Address points go to **County GIS**, which reads CSV files with latitude and longitude as well.

This manual

In the web interface: `/help`

This manual is built into bitdispatcher, so it is there wherever it runs, with no internet. Every screen's Help link opens its chapter; the palette (`Ctrl K`) finds chapters by name. The same chapters are printed as docs/manual/index.html and a PDF by `make manual` .

bitdispatcher is a product of BioDream LLC, published under the Apache License 2.0. The same chapters are under **Help** in every bitdispatcher. Map data © OpenStreetMap contributors (ODbL).