Herald

محتوا

هر هشدار به مقصد درست می‌رسد — بدون نیاز به دستور زبان دستوری.

خلاصه

رویدادهای سیستمی‌را دریافت و به‌طور قابل اعتماد به چندین کانال اعلان توزیع می‌کند تا Herald قصد آن‌هارا Herald هر هشدار به جای درست برسد. مشترکین با زبان طبیعی ساده تعامل می‌کنند؛ بازگشت → LLM از طریق یک سازوکار سه‌لایه تشخیص می‌دهد (مسیر سریع دستور → استنتاج قصد به شفاف‌سازی).

توضیح کوتاه

رویدادهای سیستمی‌را به‌طور Herald .سیستمی برای دریافت رویداد و توزیع اعلان به چندین کانال قابل اعتماد به مقصدهای درست در کانال‌های پیام‌رسان هدایت می‌کند و به مشترکین اجازه می‌دهد با و بازگشت LLM زبان طبیعی ساده صحبت کنند — قصد آن‌ها از طریق مسیر سریع دستور، استنتاج به پرسش برای شفاف‌سازی تشخیص داده می‌شود.

توضیح بلند

ستون فقرات اعلان‌ها است که تضمین می‌کند یک رویداد سیستمی واقعاً به جایی برسد که Herald یک انسان بتواند بر اساس آن اقدام کند — لایه‌ای نه چندان جذاب اما حیاتی که بیشتر سیستم‌های هشدار خانگی در آن بی‌صدا شکست می‌خورند. این سیستم رویدادهارا دریافت و به‌طور قابل اعتماد به چندین کانال اعلان توزیع می‌کند و از حالت‌های شکست آشنا جلوگیری می‌کند: جایی که یک هشدار گم می‌شود، به کانال مرده هدایت می‌شود یا در میان نویز دفن می‌شود تا زمانی که دیگر دیر شده باشد.
اما تحویل قابل اعتماد تنها نیمی از ماجرا است؛ نیمه دیگر به اتفاقی مربوط می‌شود که وقتی یک از معامله معمولی که کاربران باید دستور Herald انسان می‌خواهد پاسخ دهد رخ می‌دهد. در اینجا زبان دستوری سخت‌را برای تعامل با ربات هشدار به خاطر بسپارند سر باز می‌زند. مشترکین به‌سادگی با از طریق یک سازوکار سه‌لایه حساب‌شده منظور آن‌هارا Herald زبان طبیعی ساده می‌نویسند و تشخیص می‌دهد: مسیری سریع که دستورات صریح‌را فوراً شناسایی می‌کند، سپس استنتاج قصد مبتنی برای پیام‌های آزاد، و در نهایت بازگشت به شفاف‌سازی که وقتی (Claude از طریق کد) LLM بر → قصد واقعاً مبهم است، پاسخ می‌دهد، برچسب می‌زند و سؤال می‌پرسد. این نردبان «شناسایی استنتاج → شفاف‌سازی» کل فلسفه طراحی‌را در یک نگاه نشان می‌دهد — حالت‌رایج فوراً و قطعی باقی می‌ماند، حالت انعطاف‌پذیر توسط یک مدل مدیریت می‌شود و حالت نامطمئن هرگز با حدس کورکورانه‌ای که منجر به اقدام اشتباه شود حل نمی‌شود.

همچنین مشارکت و انتساب‌را مدل‌سازی می‌کند: یک متغیر محیطی نام کاربری اپراتور Herald (HERALD_<CHANNEL>_OPERATOR_USERNAME) و یک قرارداد مشارکت/انتساب، فیلدهای created_by/assigned_to و برچسب‌گذاری اعلان با @را هدایت می‌کند تا مشخص باشد چه ماژول Herald ،کسی چه کاری انجام داده و به چه کسی اطلاع‌رسانی می‌شود. از نظر حاکمیت را به‌عنوان زیرماژول هم‌مکان به ارث می‌برد و قوانین آن‌را دنبال می‌کند و Helix Constitution است — کل پیکره ۶۶ سندیش از مارک‌داون به Docs Chain یکی از مصرف‌کنندگان اولیه تولید متصل شده و به‌طور Docs Chain سیستم exec: از طریق تبدیل‌های HTML/PDF/DOCX با مشخصات لایه‌ای (جایگزینی Shell/Go عمدتاً ابزاری از نوع Herald .تمیز تأیید می‌شود وراهنماهای‌راه‌اندازی اپراتور برای هر کانال در پیام‌رسان‌ها و ارسال‌کننده‌های (V1→V2→V3→V4 است LLM/agent.

چرا آن‌را ساختیم

هشدارها بی‌صدا شکست می‌خورند — به کانال اشتباه ارسال می‌شوند، گم می‌شوند یا به دستور زبان برای تضمین توزیع قابل Herald .دستوری سختی نیاز دارند که کاربران آن‌را به خاطر نمی‌سپارند اعتماد و امکان پاسخگویی با زبان طبیعی ساخته شد تا اعلان‌ها هم قابل اعتماد باشند و هم اقدام بر اساس آن‌ها آسان.

محتوا

چرا این یک تغییردهندهٔ بازی است

این سیستم دو چیزی‌را که معمولاً به‌عنوان محصولات جداگانه خریداری می‌شوند — مسیریابی چندکاناله و قابل‌اعتماد رویدادها و یک‌رابط زبان طبیعی — در یک سامانه ادغام می‌کند که در آن اپراتورها صرفاً صحبت می‌کنند و نرم‌افزار منظورشان‌را درک می‌کند. جزئیاتی که آن‌را در محیط عملیاتی قابل‌اعتماد می‌سازد، سازوکار هشداردهی است: سیستمی که ترجیح می‌دهد بپرسد تا اینکه به‌اشتباه فعال شود، سیستمی است که واقعاً می‌توان به آن اجازه داد با داده‌های واقعی کار کند.

نوآوری‌ها

- شفاف‌سازی و پرسش → LLM نظم سه‌لایه‌ای قصد: مسیر سریع فرمان → استنتاج.
- تعامل مشترکین با زبان طبیعی (بدون نیاز به یادگیری نحو فرمان‌ها).
- و تگ‌گذاری با @را هدایت created_by/assigned_to قرارداد انتساب مشارکت‌کننده که می‌کند.
- مصرف‌کننده واقعی Docs Chain (مجموعه‌ای از ۶۶ سند، چندفرمت، تأییدشده).

چالش‌ها وراه‌حل‌ها

- **ابهام در قصد زبان طبیعی:** با نردبان سه‌مرحله‌ای شناسایی/استنتاج/شفاف‌سازی حل شده است، نه با حدس کورکورانه.
- **توزیع مطمئن:** با طراحی دریافت→توزیع چندکاناله حل شده تا هشدارها به مقصد صحیح برسند.
- **انتساب صحیح در کانال‌های مختلف:** با متغیر محیطی نام کاربری اپراتور و قرارداد انتساب مشارکت‌کننده حل شده است.

- **انحراف مستندات:** با تبدیل‌های تأییدشده حل شده Docs Chain با اتصال مستندات از طریق است.

پشتهٔ فناوری (چرایی و چگونگی)

- **Go** — منطق اصلی رویداد/توزیع (بر اساس الگوهای زبانی هر سازمان).
- **Shell** — ابزارها و اسکریپت‌های‌راه‌اندازی اپراتورها.
- **Claude (LLM)** کد — لایهٔ استنتاج قصد برای پیام‌های آزاد.
- **آداپتورهای کانال پیام‌رسان** — توزیع چندکاناله اعلان‌ها.
- **Docs Chain** — خط لولهٔ ساخت/تأیید مستندات (Markdown→HTML/PDF/DOCX).
- **Helix Constitution** زیرماژول — حاکمیت/قوانین به ارث رسیده.