RFI-IRFOS
Research Focus Institute · Graz

WIRKUNGSBERICHT · VEREINSJAHR 2025/26

Forschung, die allen gehört.

Offen, überprüfbar und mit Menschen an Bord. Was das Research Focus Institute zwischen dem 13.11.2025 und dem 16.11.2026 gebaut, gemessen und veröffentlicht hat.

331 Mio.
Parameter in unserem Sprachmodell albert.

6
Veröffentlichungen mit DOI

292
spezialisierte Agenten

41
offene Repositories

387+
öffentliche Apps geprüft

Wirkungsbericht nach dem Social Reporting Standard, erstellt von den Mitgliedern des Vereins.
Eingereicht im Rahmen unserer Bewerbung um das aws-Label Verified Social Enterprise am 16.11.2026.
Research Focus Institute – Interdisciplinary Research Facility for Open Sciences – Verein für Forschung · ZVR 1015608684 ·
Elisabethinergasse 25/10, 8020 Graz · rfi-irfos.com

ZUSAMMENFASSUNG

Executive Summary

Das Research Focus Institute (RFI-IRFOS) in Graz ist ein Verein, der KI-gestützte Zukunftstechnologien erforscht und die Ergebnisse veröffentlicht. Dieser Bericht folgt dem Social Reporting Standard und deckt das Vereinsjahr 2025/26 ab (13.11.2025 bis 16.11.2026).

Ergebnis

Im Berichtszeitraum entstanden ein ternäres Sprachmodell mit 331 Mio. Parametern, ein von Laura Serna Gaviria gebautes Team aus 292 spezialisierten Agenten, 41 öffentliche Repositories mit 11.269 Commits und sechs Veröffentlichungen mit DOI. In der Sicherheitsforschung haben wir 387+ öffentliche Apps auf Quellcode-Ebene geprüft und 249 Berichte veröffentlicht, die Betreiber vor der Veröffentlichung erhalten.

Wirkung

Kleine Unternehmen bekommen Zugang zu Sicherheitsprüfung und KI, die bisher großen Organisationen vorbehalten waren. Ternäres Rechnen senkt dafür die Einstiegshürde, denn auf einer CPU von 2013 braucht es 4× weniger Speicher. Das Jahr trägt zu den UN-Zielen 8, 9 und 10 bei (Kapitel 2.3.3).

Ausblick

Wir bauen unsere Server so weit aus, dass wir das Sprachmodell lokal trainieren und die Forschung zur ternären Logik skalieren können. Zahlende Kunden des Managed SOC finanzieren den Ausbau, das Ziel sind fünf bis zum 31.12.2026. Die Wirkung bei Kunden messen wir ab 2027 (Kapitel 4.1).

AUF EINEN BLICK

Dieses Jahr in Zahlen

Heute legen wir offen, was im Vereinsjahr entstanden ist. Die wichtigsten Zahlen stehen hier, jede mit ihrer Quelle in Kapitel 3 und Anhang A.

331 Mio.
Parameter im Sprachmodell albert.
ternäres Modell, jedes Gewicht −1, 0 oder +1

6
Veröffentlichungen mit DOI
frei lesbar auf OSF

292
spezialisierte Agenten
in 39 Bereichen eingeteilt

4×
weniger Speicher
gemessen auf einer CPU von 2013

84
Token pro Sekunde
albert. auf derselben CPU

249
veröffentlichte Berichte
als PDF auf rfi-irfos.com

11.269
Commits
im Hauptzweig seit 13.11.2025

41
öffentliche Repositories
27 davon mit freier Lizenz

15.337
Downloads auf crates.io
über mindestens 51 Pakete

387+
öffentliche Apps geprüft
auf Quellcode-Ebene zerlegt

1,04 Mio.+
Klassen gelesen
der ausgelieferte Code, gezählt

262+
kritische Befunde
in Kopie an Aufsichtsbehörden

Offen

Forschungsberichte, Messwerte und Software stehen öffentlich. Wer sie prüfen will, findet Befehle zum Nachmessen und Zeilennummern im Benchmark-Dokument.

Gemessen

Auf einer CPU von 2013 braucht ternäres Rechnen 4× weniger Speicher und liefert 1,63× mehr Durchsatz. Jede Zahl trägt ein Etikett: gemessen, belegt oder Hypothese.

Mit Menschen

Agenten finden, Menschen entscheiden. Das Managed SOC setzt keinen Compliance-Punkt allein durch einen Scan auf „erfüllt“.

Grenzen

Wo wir keine Zahl haben, sagen wir es. Kundenwirkung messen wir ab 2027.

Stand 16.11.2026. Das Modell albert. ist ein Forschungsprototyp.

INHALT

Inhaltsverzeichnis

1 **Terminologie** 5

2 **Das gesellschaftliche Problem und unser Lösungsansatz** 5
- 2.1 Das gesellschaftliche Problem 5
- 2.2 Bisherige Lösungsansätze 7
- 2.3 Unser Lösungsansatz 7

3 **Ressourcen, Leistungen und Wirkungen im Berichtszeitraum** 10
- 3.1 Eingesetzte Ressourcen (Input) 10
- 3.2 Erbrachte Leistungen (Output) 10
- 3.3 Erreichte Wirkungen (Outcome und Impact) 14
- 3.4 Darstellung der Ressourcen, Leistungen und Wirkungen 14
- 3.5 Evaluation und Qualitätssicherung 15
- 3.6 Zielerreichung und Lernerfahrungen 15
- 3.7 Das Jahr in der Versionsverwaltung: Arbeitsnachweis 15
- 3.8 Whitebox: nachvollziehbare KI und unsere Sicherheitsrahmen 18
- 3.9 Fallstudien aus dem Berichtszeitraum 25
- 3.10 Alle Repositories im Überblick: vom Fundament bis zur Anwendung 34
- 3.11 Veröffentlichte Papers 39

4 **Planung und Ausblick** 44
- 4.1 Planung und Ziele 44
- 4.2 Chancen und Risiken 44

5 **Organisationsstruktur und Team** 46
- 5.1 Organisationsstruktur 46
- 5.2 Vorstellung der handelnden Personen 46
- 5.3 Partnerschaften, Kooperationen und Netzwerke 47

6 **Forschungspartnerschaft mit Laura Serna Gaviria** 48
- 6.1 Rolle und Zusammenarbeit 48
- 6.2 Was im Vereinsjahr auf Lauras Arbeit beruht 48

Anhang A: Zählweise und Grenzen 49

Anhang B: Wörtliche Beispiele aus den Commit-Nachrichten 50

Quellenverzeichnis 53

KAPITEL 1

Terminologie

Wirkungen (Outcome und Impact) sind Veränderungen bei Zielgruppen, in ihrem Umfeld und in der Gesellschaft, die auf unsere Leistungen folgen. Wir stellen sie in der IOOI-Wirkungskette dar: **Ressourcen (Input) → Leistungen (Output) → Wirkungen (Outcome und Impact)**.

Abkürzungen: KI = künstliche Intelligenz, KMU = kleine und mittlere Unternehmen, SOC = Security Operations Center (Sicherheitszentrale, die Sicherheitsvorfälle laufend beobachtet und bearbeitet), SDG = Ziel für nachhaltige Entwicklung der Vereinten Nationen, TWh = Terawattstunde.

Aussagen in diesem Bericht gehören einer von drei Arten an: **gemessen** (eigene Messung, mit Quelle), **belegt** (externe Quelle, im Quellenverzeichnis) oder **geplant** bzw. **Hypothese** (so gekennzeichnet).

KAPITEL 2

Das gesellschaftliche Problem und unser Lösungsansatz

Kleine Betriebe prüfen ihre Software selten, KI-Rechenleistung sammelt sich bei wenigen Anbietern, und viele Fachleute sorgen sich um ihre Arbeit. Dieses Kapitel belegt die drei Probleme mit Studien und zeigt, wie wir an jedem ansetzen.

2.1 Das gesellschaftliche Problem

Gesellschaftliche Ausgangslage

Fortschrittliche KI und wirksame Software-Sicherheit stehen vor allem großen Organisationen zur Verfügung. Wer klein ist, hat weder ein eigenes Sicherheitsteam noch die Mittel für Rechenleistung und Fachkräfte. Gleichzeitig verändert KI die Arbeit vieler Menschen, und die Unsicherheit darüber wächst. Wir sehen drei zusammenhängende Probleme.

Problem 1: Kleine Betriebe bleiben bei der Sicherheit ungeprüft.

- Nach EY Cybersecurity 2025 hatten **32 %** der befragten österreichischen Unternehmen in den letzten fünf Jahren konkrete Hinweise auf Angriffe. Phishing ist mit **73 %** die häufigste Angriffsart. Befragt wurden 200 Führungskräfte aus Unternehmen **ab 20 Beschäftigten**.
- Unternehmen mit weniger als 20 Beschäftigten kommen in dieser Studie nicht vor. Für Kleinstunternehmen (bis zu 9 Beschäftigte nach EU-Definition) gibt es dadurch kaum belastbare Zahlen. Das gehört zum Problem: Was niemand misst, wird selten behandelt.

- Fast alle Unternehmen in Österreich sind klein: Eine Schätzung von KMU Forschung Austria nennt für 2022 rund **601.300** KMU, das sind **99,8 %** der marktorientierten Unternehmen.
- Es fehlen Fachkräfte mit den richtigen Fähigkeiten: **88 %** von 16.029 befragten Fachleuten berichten in der ISC2-Studie 2025 von mindestens einer bedeutenden Sicherheitsfolge durch fehlende Fähigkeiten im Team.

Problem 2: Die Sorge um Arbeit durch KI. Der Weltwirtschaftsforum-Bericht „Future of Jobs 2025" erwartet bis 2030 **170 Mio.** neue und **92 Mio.** wegfallende Stellen (netto **+78 Mio.**), getrieben von Technologie, Demografie und weiteren Entwicklungen. **41 %** der befragten Arbeitgeber planen, ihre Belegschaft zu verkleinern, weil KI Aufgaben automatisiert. Von 100 Beschäftigten brauchen laut Bericht 59 neue oder erweiterte Fähigkeiten bis 2030, 11 davon erhalten sie voraussichtlich nicht.

Problem 3: Hoher Rechenaufwand schließt viele aus. Rechenzentren verbrauchten 2024 weltweit **415 TWh** Strom (rund 1,5 % des weltweiten Verbrauchs). Die Internationale Energieagentur erwartet bis 2030 rund **945 TWh**, mehr als das Doppelte. *Hypothese:* Je mehr Rechenleistung ein Verfahren braucht, desto weniger Organisationen können es selbst betreiben. Das verstärkt die Abhängigkeit von wenigen großen Anbietern.

Zwei Entwicklungen, ein Zugangsproblem

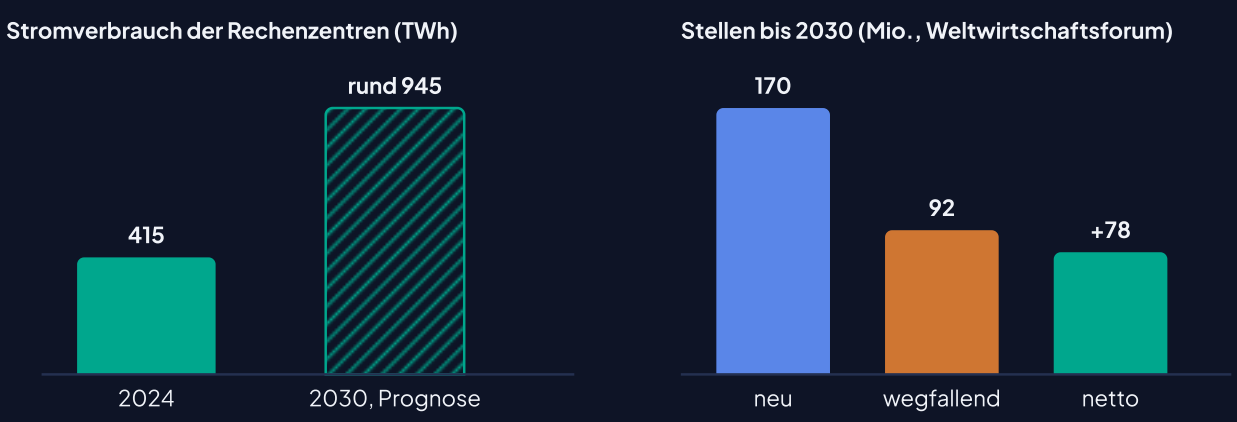


Abbildung 1: Quellen: Internationale Energieagentur (IEA 2025) und Weltwirtschaftsforum, Future of Jobs 2025. Schraffiert ist eine Prognose.

Ursachen und Folgen (Ursachenkette)

Ursache	→	Verteilung	→	Folge für Kleine	→	Folge für Menschen
Leistungsfähige KI und Sicherheitsprüfung brauchen viel Rechenleistung und seltenes Fachwissen		Beides sammelt sich bei großen Organisationen und wenigen Anbietern		Software bleibt ungeprüft, Angriffe treffen ungeschützte Betriebe		Sorge um Arbeit, wenig Einblick, wie KI zu Ergebnissen kommt

Abbildung 2: Ursachenkette. Quellen: EY Cybersecurity 2025, ISC2 2025, Weltwirtschaftsforum 2025, IEA 2025 (siehe Quellenverzeichnis). Die Verbindung der drei Probleme ist unsere Hypothese.

Betroffene Zielgruppen

- Startups, kleine und mittlere Unternehmen und Kleinstbetriebe mit eigener Software, Website oder Plattform, vom Konzern bis zum kleinen Friseursalon mit Buchungssystem.
- Sicherheits- und IT-Fachleute, deren Arbeit sich durch KI verändert.

- Forschende, Entwickler:innen und Gemeinschaften, die offene, nachprüfbare Technik brauchen.

2.2 Bisherige Lösungsansätze

Öffentliche Stellen und Verbände stellen Leitfäden und Informationen zur Cybersicherheit bereit, während kommerzielle Anbieter Penetrationstests und Managed-Security-Dienste anbieten. Außerdem finden offene Werkzeuge wie Semgrep, Gitleaks und Trivy Schwachstellen im Code.

Nach unserer Einschätzung (Hypothese) reichen diese Ansätze für kleine Betriebe nicht aus: Leitfäden prüfen keine konkrete Software, kommerzielle Dienste richten sich überwiegend an Organisationen mit Sicherheitsabteilung oder Budget, und offene Werkzeuge liefern Befundlisten, die jemand mit Fachwissen bewerten und in Lösungen übersetzen muss. Hinzu kommt, dass KI-Dienste großer Anbieter meist als geschlossene Systeme arbeiten, deren Ergebnisse sich schwer nachvollziehen lassen.

2.3 Unser Lösungsansatz

RFI-IRFOS ist ein interdisziplinäres Forschungsinstitut mit mehreren Forschungsfeldern und arbeitet nach den Prinzipien der offenen Wissenschaft (Open Science). Zweck des Vereins laut Statuten (2.1): die Erforschung von KI-gestützten Zukunftstechnologien. Die Tätigkeit ist nicht auf Gewinn gerichtet (2.2). Überschüsse werden bis zum Monatsende reinvestiert, wobei höchstens 10 % als Rücklage bleiben dürfen, und zum Jahresende wird der gesamte Überschuss bis auf null ausgegeben, also reinvestiert. Damit finanzieren sie künftige Vereinszwecke (3.2).

Wir setzen an allen drei Problemen an, mit einem Grundsatz: **Wissen offen machen, Aufwand senken, Menschen einbinden.**

- **Offen machen:** Forschungsberichte, Messwerte und Software veröffentlichen wir, damit andere sie prüfen und nutzen können.
- **Aufwand senken:** Unsere Forschung zur ternären Logik (Rechnen mit drei Zuständen: ja, nein, nichts tun) senkt Speicher- und Rechenbedarf. Das erlaubt leistungsfähige KI auf Standard-Hardware.
- **Menschen einbinden:** Unsere Produkte arbeiten nach dem Prinzip „Agenten finden, Menschen entscheiden“. Das Ergebnis bleibt nachvollziehbar (Whitebox-Ansatz, Kapitel 3.8).

Aus der Forschung entstehen kommerzielle Leistungen, die über das Gewerbe des Vereins „Dienstleistungen in der automatischen Datenverarbeitung und Informationstechnik“ laufen und die Forschung finanzieren.

2.3.1 Leistungen (Output) und direkte Zielgruppen

LEISTUNG	INHALT	DIREKTE ZIELGRUPPE
Offene Forschung	Veröffentlichte Untersuchungsberichte (Disclosure Reports) zu Apps im Auftrag von Forschungspartnern. Jeder Bericht verbindet Technik, Datenschutzrecht (DSGVO Artikel für Artikel), Verbraucherschutz und Aufsichtsrecht in einer Analyse. Betreiber erhalten den Bericht vor der Veröffentlichung, bestätigte Befunde gehen in Kopie auch an Aufsichtsbehörden.	Forschungspartner, Aufsicht, Fachöffentlichkeit

Offene Software und Messwerte	Öffentliche Code-Repositories mit freien Lizenzen, darunter eine eigene Systemprogrammiersprache (Ternlang) und ein eigenes Betriebssystem (Rusty Penguin). Ein öffentliches Benchmark-Dokument mit nachprüfbaren Messwerten zur ternären Logik.	Entwickler:innen, Forschende, Open-Source-Gemeinschaft
Sicherheitsuntersuchung mit Menschen an Bord (The Squad)	Eigens entwickelte KI-Agenten untersuchen tausende Angriffswege parallel. Erfahrene Spezialisten bewerten die Befunde und entwickeln gezielte Lösungen.	Startups, KMU, Kleinstbetriebe
Managed SOC	Ein Ort für Befunde, Status und Compliance-Nachweise. Anbindung von Code-Repositories, Zuordnung der Befunde zu Regelwerken, Weg von der Aufgabe bis zur Korrektur (Pull Request) beim Kunden.	Startups, KMU, Kleinstbetriebe
Whitebox und Sicherheitsrahmen	Nachvollziehbare KI-Werkzeuge: Beleg-Status je Aussage, erklärte Zahlen, verkettete Protokolle, menschliche Freigabe vor Handlungen, Schutzschichten für Agenten (Kapitel 3.8). Teile davon sind öffentlich, etwa Entscheidungsprüfer und Sicherheitstore in Ternlang.	Alle Zielgruppen, besonders Betreiber und Aufsichtsbehörden
Wissensvermittlung	Vorträge und Workshops zu KI und Sicherheit.	Unternehmen, Fachpublikum

Entgelt: Forschungsberichte, Software und Messwerte stehen offen zur Verfügung. Die Produkte sind entgeltlich, das Managed SOC ab 99 € pro Monat. Das Entgelt zahlen die Kunden. Es finanziert Betrieb und Forschung.

2.3.2 Intendierte Wirkungen auf direkte und indirekte Zielgruppen

- **Kleine Betriebe:** Software wird geprüft, die sonst ungeprüft bliebe. Die Plattform skaliert vom Konzern bis zum kleinen Betrieb: dieselben Agenten, dasselbe Portal, passend zur Größe. *Indirekt:* Kundinnen und Kunden dieser Betriebe, deren Daten sicherer sind.
- **Fachleute:** Agenten schaffen Kapazität, Menschen bewerten und entscheiden. Fachleute behalten ihre Arbeit und gewinnen Zeit für schwierige Fälle. Im Portal gilt: Ein automatisch geprüfter Punkt gilt nicht als erfüllt, die Entscheidung trifft ein Mensch.
- **Forschung und Gemeinschaft:** Offene Ergebnisse lassen sich prüfen, wiederverwenden und weiterentwickeln.
- **Gesellschaft und Umwelt:** Geringerer Speicher- und Rechenbedarf senkt die Einstiegshürde für KI, denn wer weniger Hardware braucht, kann KI selbst betreiben und ist auf gemietete Rechenleistung weniger angewiesen.

2.3.3 Darstellung der Wirkungslogik

ZIELGRUPPE	LEISTUNG	ERWARTETE WIRKUNG
Startups, KMU, Kleinstbetriebe	Squad-Untersuchung, Managed SOC, automatisierte Compliance-Zuordnung	Zugang zu Sicherheitsprüfung ohne eigene Abteilung; weniger Ungleichheit zwischen großen und kleinen Betrieben (SDG 10)
Sicherheits- und IT-Fachleute	Agenten liefern Kapazität, Menschen entscheiden	Arbeitsplätze bleiben, Zeit für schwierige Fälle (SDG 8)
Forschende, Entwickler:innen	Offene Berichte, Software, Benchmarks	Nachprüfbares, wiederverwendbares Wissen (SDG 9)
Gesellschaft, Umwelt	Ternäre Effizienz auf Standard-Hardware	Geringerer Speicher- und Rechenbedarf, niedrigere Einstiegshürde für KI (SDG 9)
Verein	Überschüsse werden monatlich reinvestiert	Dauerhafte Finanzierung weiterer Forschung (Impact First)

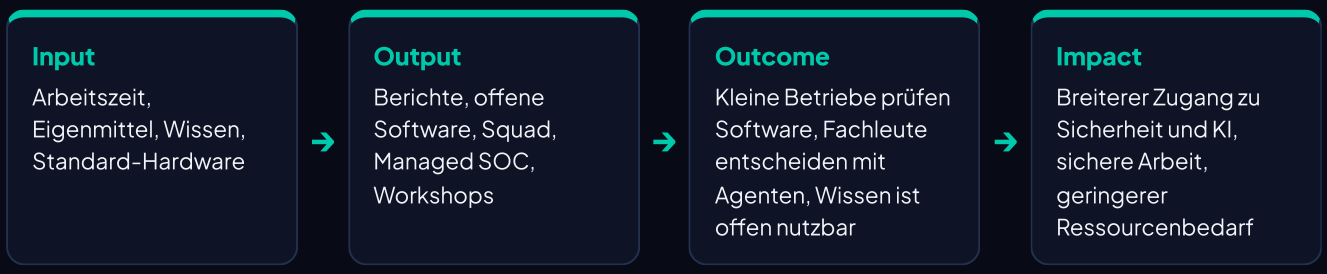
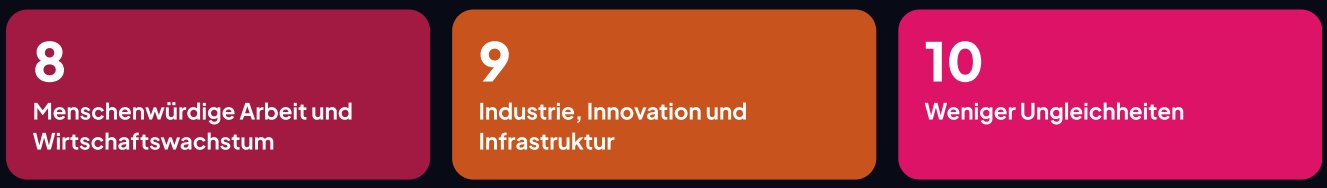


Abbildung 3: Wirkungskette von RFI-IRFOS.

Beitrag zu den Zielen für nachhaltige Entwicklung



SDG	BEGRÜNDUNG (BELEGBAR)
8 Menschenwürdige Arbeit und Wirtschaftswachstum	Agenten ergänzen Fachleute und Menschen entscheiden (Kapitel 2.3.2), während die Produkte die Forschung finanzieren (Kapitel 3.1).
9 Industrie, Innovation und Infrastruktur	Eigene Forschung und Software, öffentlich nachprüfbar. Ternäre Verfahren mit gemessenem Speicher- und Rechenvorteil (Kapitel 3.2).
10 Weniger Ungleichheiten	Dieselbe Technik für den Konzern und den kleinen Betrieb, passend zur Größe der Kundschaft (Kapitel 2.3.2).

KAPITEL 3

Ressourcen, Leistungen und Wirkungen im Berichtszeitraum

Hier steht, was im Vereinsjahr entstanden ist, von der ersten Zeile Code bis zum Kundenportal. Jede Kennzahl nennt Quelle und Abrufdatum. Die Fallstudien erzählen auch, was schiefging und wie wir es korrigiert haben.

Berichtszeitraum ist das Vereinsjahr 2025/26 (13.11.2025 bis 16.11.2026). Die Zahlen beziehen sich auf den Stand am 16.11.2026. Der Verein führt die Arbeit der Gründergruppe fort, die seit 2020 gemeinsam forscht (siehe Kapitel 5.1). Die Kennzahlen zur Forschung sind daher kumuliert.

3.1 Eingesetzte Ressourcen (Input)

- **Zeit:** Die Arbeitszeit des Kernteams und der Mitglieder.
- **Finanzielle Mittel:** Der Verein finanziert sich vollständig selbst, aus Vorträgen, Workshops, Spenden und viel Eigenarbeit.
- **Sachmittel:** Eigene Rechner. Alle Benchmarks laufen auf Standard-Hardware (CPU eines Laptops aus dem Jahr 2013), ohne Grafikkarte.
- **Immaterielle Ressourcen:** Wissen und Fähigkeiten des Teams, eigene Software unter freien Lizenzen (13 Repositories Apache-2.0, 8 MIT, 2 LGPL, 4 weitere), Forschungspartnerschaften.

3.2 Erbrachte Leistungen (Output)

Veröffentlichte Forschung (Stand 16.11.2026, Quelle: rfi-irfos.com, Abschnitt „Our research, published“)

INDIKATOR	WERT	BEDEUTUNG
Mobile Apps geprüft	**387+**	Jede App auf Quellcode-Ebene zerlegt und untersucht, nicht nur von außen gescannt
Gelesene Klassen	**1,04 Mio.+**	Der ausgelieferte Code, gelesen und gezählt
Kritische Befunde	**262+**	Bestätigte Befunde, in Kopie an Aufsichtsbehörden
Schwachstellen	**970+**	Im Programmcode nachgewiesene Tracker von Drittanbietern
Untersuchte Endpunkte	**567+**	Server und Länder, in die Daten fließen

Kartierte SDK-Instanzen	132+	Wiederverwendbare Bibliotheken, über den gesamten Bestand verknüpft

Veröffentlichte Untersuchungsberichte: 249 Berichte als PDF auf rfi-irfos.com (Zählung am 16.11.2026). Jeder Bericht enthält Zusammenfassung, Befunde mit Behebungsvorschlag, Wirkungsanalyse, technische Belege, Zeitleiste der Offenlegung und Restrisiko mit Empfehlungen.

Offene Software (GitHub-Organisation rfi-irfos, Stand 16.11.2026, gemessen über die GitHub-Schnittstelle)

INDIKATOR	WERT
Repositories gesamt	**59 (41 öffentlich, 18 privat)**
Öffentliche Repositories ohne Fork und Archiv	**23**
Öffentliche Repositories mit freier Lizenz	**27 (Apache-2.0: 13, MIT: 8, LGPL: 2, weitere: 4)**
Sterne auf den 23 eigenständigen öffentlichen Repositories	**68**
Commits im Forschungs-Repository (ternary-intelligence-stack, 01.04. bis 13.11.2026)	**4.705**
Commits in allen eigenen Repositories seit 13.11.2025 (Hauptzweig, Details in Kapitel 3.7)	**11.269**

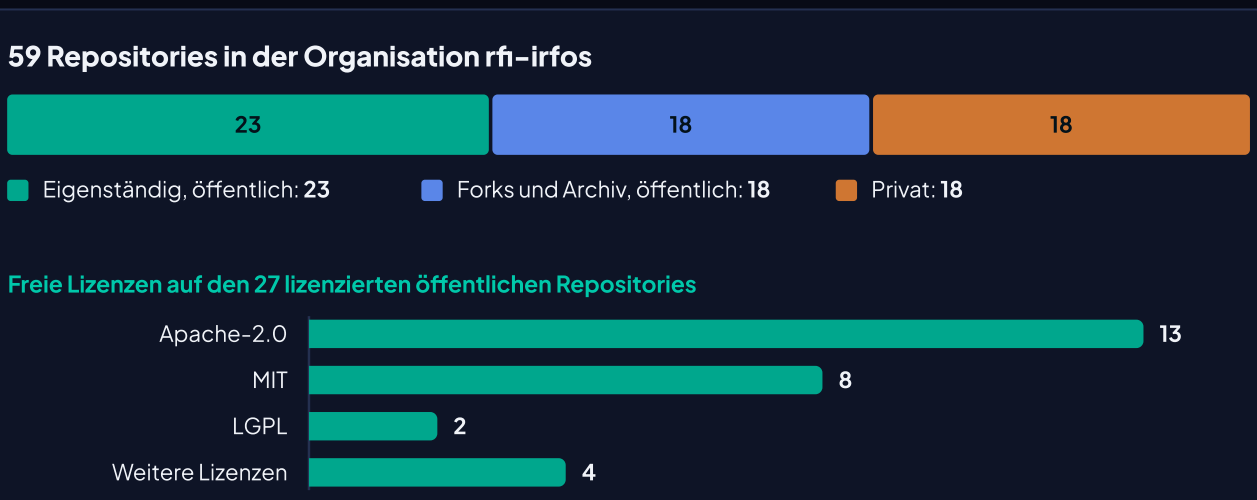


Abbildung 4: Zählung über die GitHub-Schnittstelle am 16.11.2026. Die 27 Lizenzen umfassen auch lizenzierte Forks.

Die zwei wirkungsstärksten Projekte

PROJEKT	WAS ES IST	REICHWEITE (STAND 16.11.2026)

Ternary Intelligence Stack mit Ternlang LGPL-3.0, Rust	Eine eigene Systemprogrammiersprache mit Compiler und Laufzeitumgebung auf Basis balancierter ternärer Logik. Der Grundtyp *trit* kennt drei Werte: –1 (ablehnen), 0 (halten), +1 (zustimmen). Dazu eine Standardbibliothek, ein KI-Trainings- und Inferenz-Rahmenwerk und ein Server für KI-Werkzeuge (MCP).	30 Sterne; 20 Versionsmarken; auf crates.io veröffentlichte Pakete (siehe unten)
Rusty Penguin MIT, Rust	Ein Betriebssystem, komplett von Grund auf in Rust geschrieben: eigener Bootloader, Kernel, Treiber, Fenstersystem und Anwendungen, ohne Linux-Kernel und ohne C-Standardbibliothek. Ternäre Logik ist in jeder Schicht ein Grundbaustein. Aktive Entwicklung, Version 3.8.0.	18 Sterne; 4 Versionsmarken; offenes Entwicklungsprotokoll (SESSION_LOG)

Beide Projekte weisen offen aus, dass ein Teil des Codes mit KI-Unterstützung entsteht: Menschen entwerfen und prüfen, jeder Meilenstein wird in einer virtuellen Maschine (QEMU) verifiziert, bevor wir ihn als erreicht ausweisen. Quellen: README-Dateien der Repositories, crates.io (Abruf 16.11.2026).

Reichweite der Software (Stand 16.11.2026, Primärquellen)

KANAL	WERT	HINWEIS
crates.io (Paketregister für Rust)	**mindestens 51 Pakete, 15.337 Downloads gesamt**	Größte Pakete: ternlang-core 2.675, ternlang-ml 1.943, albert-runtime 1.166, ternlang-moe 1.086.
HuggingFace (Modelle und Daten)	**3 Modelle, 1 Datensatz**	Modell albert.: 261 Downloads, 2 Likes (laut Anzeige von HuggingFace). Datensatz mit Ursachenketten: 49 Downloads.
GitHub-Organisation	**78 Sterne**	71 davon auf öffentlichen Repositories (Zählung über die GitHub-Schnittstelle).
Website rfi-irfos.com	**249 Berichte, 387 Verzeichniseinträge**	Kennzahlen live auf der Seite einsehbar.

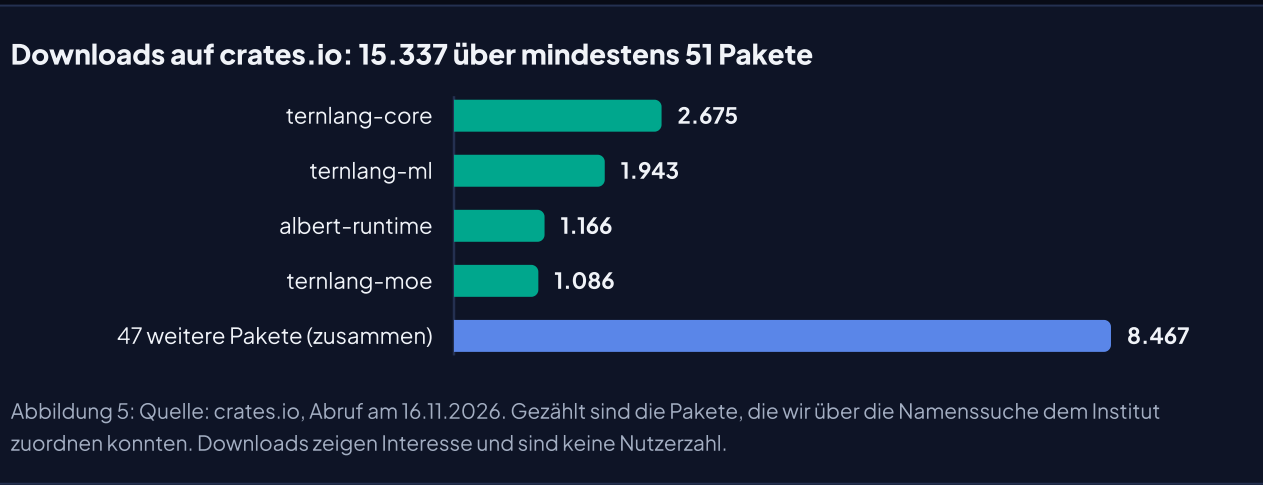


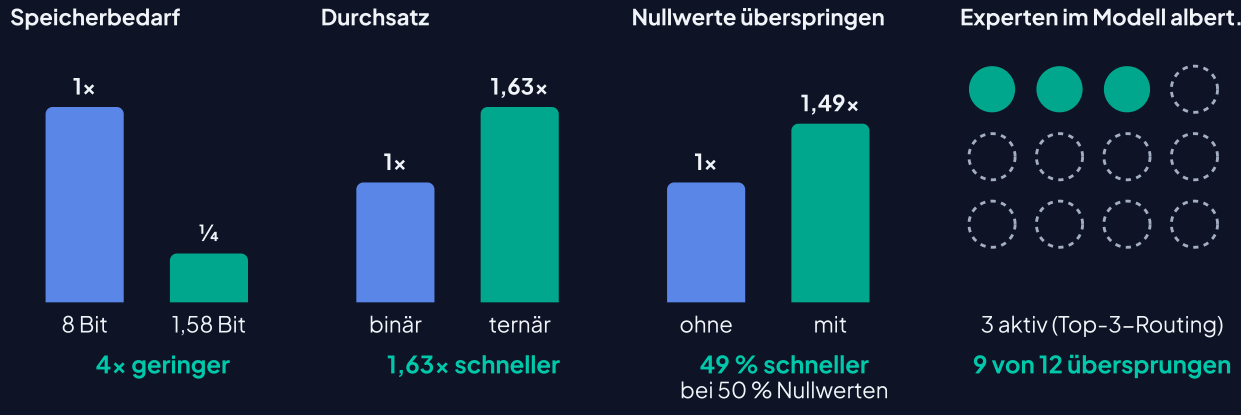
Abbildung 5: Quelle: crates.io, Abruf am 16.11.2026. Gezählt sind die Pakete, die wir über die Namenssuche dem Institut zuordnen konnten. Downloads zeigen Interesse und sind keine Nutzerzahl.

Interne Messung. Zusätzlich messen wir die Reichweite in einem eigenen Dashboard, das Zahlen über alle Repositories und Kanäle aufsummiert: rund 124 Sterne, 130.300 Klone von Repositories, 24 Beobachter, 3.200 Aufrufe unseres Servers für KI-Werkzeuge bei der Registry Smithery, 2.400 Downloads einer Editor-Erweiterung (OpenVSX) und 19.100 Paket-Downloads auf crates.io. Diese Werte sind Eigenmessungen, nicht unabhängig geprüft und mit den Einzelwerten oben nicht deckungsgleich, weil sie weitere Konten und Kanäle zusammenfassen. Klone enthalten auch automatische Abrufe und gelten daher nur als Hinweis auf Interesse.

Messwerte der ternären Forschung (öffentliches Dokument BENCHMARKS.md im Repository ternary-intelligence-stack, CPU ohne Grafikkarte)

MESSWERT	ERGEBNIS	ABSCHNITT
Speicherbedarf (1,58 Bit gegen 8 Bit)	**4× geringer**	§7
Durchsatz gegen optimierte binäre Schleifen	**1,63× schneller**	§7
Überspringen von Nullwerten bei 50 % Sparsity	**49 % schneller**	§8
Übersprungene Experten im Modell albert. (Top-3-Routing)	**75 % (9 von 12)**	§11
Antwortgeschwindigkeit des Modells albert. auf einer CPU	**84 Token/s bei rund 0,009 $ Strom pro Stunde**	§16d

Ternäre Logik, gemessen auf einer CPU von 2013



84 Token/s
Antwortgeschwindigkeit von albert. auf einer CPU

rund 0,009 $
Strom je Stunde

Intel i7-4800MQ
(Laptop, 2013), keine Grafikkarte

Abbildung 6: Quelle: BENCHMARKS.md im öffentlichen Repository ternary-intelligence-stack, §7, §8, §11 und §16d.

Die Werte stammen von einem Laptop-Prozessor aus dem Jahr 2013 (Intel i7-4800MQ). Hardware-Hochrechnungen im Dokument (§15) sind als Projektion gekennzeichnet und fließen hier nicht ein. Das Modell albert. ist ein Forschungsprototyp.

Produkte

- **Managed SOC:** live auf portal.rfi-irfos.com, mit Anmeldung per Mehrfaktor-Verfahren, mandantengetrennten Daten und Einladung neuer Kunden. Anbindung von GitHub, GitLab, AWS, Azure, Google Cloud und Kubernetes. Quellcode-Analyse mit Semgrep, Gitleaks und Trivy. Kontrollkatalog mit

neun Regelwerken (ISO 27001, SOC 2, NIS2, EU CRA, NIST CSF 2.0, CIS Controls v8, DSGVO, DORA, PCI DSS 4.0) und vier bedingten Regelwerken.

- **The Squad:** Plattform für Sicherheitsuntersuchungen mit eigener Multi-Agenten-Umgebung und menschlicher Prüfung.

Wissensvermittlung: Vorträge und Workshops sowie regelmäßige Beiträge auf Social Media (u. a. LinkedIn).

3.3 Erreichte Wirkungen (Outcome und Impact)

- **Nachprüfbarkeit:** Forschungsergebnisse und Messwerte sind öffentlich und lassen sich nachvollziehen, denn Benchmarks tragen Abschnittsnummern und Repositories eine Lizenz (Befehle zum Nachmessen: Kapitel 3.5).
- **Behebung von Sicherheitsproblemen:** Betreiber erhalten jeden Bericht vor der Veröffentlichung, mit Befund, Beleg und Behebungsvorschlag je Befund. Die Veröffentlichung folgt nach einer Frist von 90 Tagen (Beispiel: gemeldet am 4. August, veröffentlicht am 2. November 2026). Der jüngste Bericht dokumentiert die Antworten des Betreibers und prüft die Korrekturen im Programmcode der neuen Version: ein Befund behoben und bestätigt, einer teilweise behoben, einer strittig.
- **Wirkung im eigenen Betrieb:** Wir haben unsere eigenen Repositories an das Managed SOC angebunden und im Einsatz erprobt: Befund, Zuweisung, Korrektur als Patch oder Pull Request, Prüfung des Merge. Das Portal dient damit als Machbarkeitsnachweis mit funktionierendem Geschäftsmodell.
- **Technische Wirkung:** Die gemessenen Speicher- und Rechenvorteile (Kapitel 3.2) zeigen, dass leistungsfähige KI auf Standard-Hardware läuft.
- **Mögliche Nebenfolge:** Ein veröffentlichter Befund könnte missbraucht werden, solange die Lücke offen ist. Dagegen schützen die Vorabinformation der Betreiber, die genannte Frist von 90 Tagen und technische Belege ohne personenbezogene Daten in den Berichten.
- **Hinweis:** Das Portal ist seit November 2026 live, und Wirkungen bei zahlenden Kunden messen wir ab dem Start der ersten Kundenprojekte (Indikatoren: Kapitel 4.1).

3.4 Darstellung der Ressourcen, Leistungen und Wirkungen

WIRKUNGSKETTE	BERICHTSZEITRAUM	INDIKATOR
Input	Arbeitszeit des Kernteams, Eigenfinanzierung aus Vorträgen, Workshops, Spenden	Kernteam und Mitglieder, eigene Standard-Hardware
Output	Veröffentlichte Forschung, offene Software, Messwerte, Portal, Squad	387+ Apps, 249 veröffentlichte Berichte, 1,04 Mio.+ Klassen, 41 öffentliche Repositories, 27 mit Lizenz, 5 Benchmark-Kernwerte, 11.269 Commits, 51 Pakete auf crates.io
Outcome	Nachprüfbare Ergebnisse, Befunde bei Aufsichtsbehörden, Portal im Einsatz	262+ kritische Befunde, Portal live mit Preismodell

Impact	Zugang zu Sicherheit und KI wird breiter, geringerer Ressourcenbedarf	4× weniger Speicher, 1,63× Durchsatz (gemessen); Kundenwirkung ab 2027 gemessen

3.5 Evaluation und Qualitätssicherung

- **Reproduzierbare Benchmarks:** Das Benchmark-Dokument enthält Befehle zum Nachmessen und ein gehärtetes Verfahren mit statistischer Prüfung (§10).
- **Menschliche Prüfung:** Befunde der Agenten bewerten Fachleute, bevor sie bei Kunden ankommen (Regel für Compliance-Punkte: Kapitel 2.3.2).
- **Überprüfung der Korrekturen:** Meldet ein Betreiber eine Korrektur, prüfen wir sie im Programmcode der neuen Version. Strittige Punkte stehen als strittig im Bericht, mit beiden Standpunkten.
- **Integrität der Daten:** Der Datenexport für das Portal prüft vor der Veröffentlichung die Kette der Ereignis-Prüfsummen. Der Export bricht ab, wenn die Prüfung scheitert.
- **Tests:** 48 automatisierte Testskripte zu Anmeldung, Mehrfaktor-Verfahren, Anbindungen und Zahlungsabwicklung.
- **Herkunftskennzeichnung:** Interne Forschungsdaten und öffentliche Bedrohungsdaten stehen im Portal getrennt und mit Quellenangabe. Es gibt keine erfundenen Demodaten im Kundenbereich.

3.6 Zielerreichung und Lernerfahrungen

Für den Berichtszeitraum 13.11.2025 bis 16.11.2026 liegen keine Vorjahresziele des Vereins vor. Drei Lernerfahrungen:

- **Wirkung nur mit Beleg behaupten.** Wir haben bewusst keine Angabe zum Energiebedarf in Watt veröffentlicht, solange keine Messung vorliegt. Veröffentlicht ist, was gemessen ist: Speicher, Durchsatz, Rechenzeit. Zahlen werden berechnet statt getippt, und Berichte durchlaufen vor der Veröffentlichung eine zweite Prüfung durch Fachleute.
- **Transparenz über den Reifegrad.** Das Portal ist jung, deshalb belegen wir seine Reife über Live-Betrieb, Tests und eigene Nutzung und können uns auf keine lange Entwicklungsgeschichte berufen.
- **Ordentlicher Übergang.** Der Wechsel vom Vorarlberger Verein zum Verein in Graz lief lückenlos: Der neue Verein entstand, der alte wurde erst nach Abschluss aller Arbeiten aufgelöst.

3.7 Das Jahr in der Versionsverwaltung: Arbeitsnachweis

Die gesamte Entwicklungsarbeit des Instituts liegt in Git-Repositories. Git ist eine Versionsverwaltung: Sie speichert jeden Änderungsschritt (Commit) mit Datum, Autor und Kurzbeschreibung. Wir haben am 16.11.2026 die vollständige Historie aller 41 eigenen Repositories der GitHub-Organisation ausgewertet (ohne übernommene Fremdprojekte). 40 davon enthalten Commits im Berichtszeitraum. Gezählt ist jeweils der Hauptzweig. Damit ist die Arbeit des Jahres Schritt für Schritt belegt, mit Datum und Wortlaut. Aussagen in diesem Kapitel stützen sich auf diese Historie und sind dort nachprüfbar.

KENNZAHL (STAND 16.11.2026)	WERT

Commits im Hauptzweig seit 13.11.2025	**11.269**
davon reine Zusammenführungen von Zweigen (ohne eigene Änderung)	**152**
inhaltliche Commits	**11.117**
Tage mit mindestens einem Commit	**222**
Repositories mit Commits (öffentlich / privat)	**40 (25 / 15, private hier nur gezählt, nicht beschrieben)**
Veröffentlichte Versionsmarken (Tags)	**Ternlang 20, Rusty Penguin 4, pass-the-bot 3, weitere 5**

Verlauf nach Monaten

11.269 Commits im Hauptzweig, nach Monaten

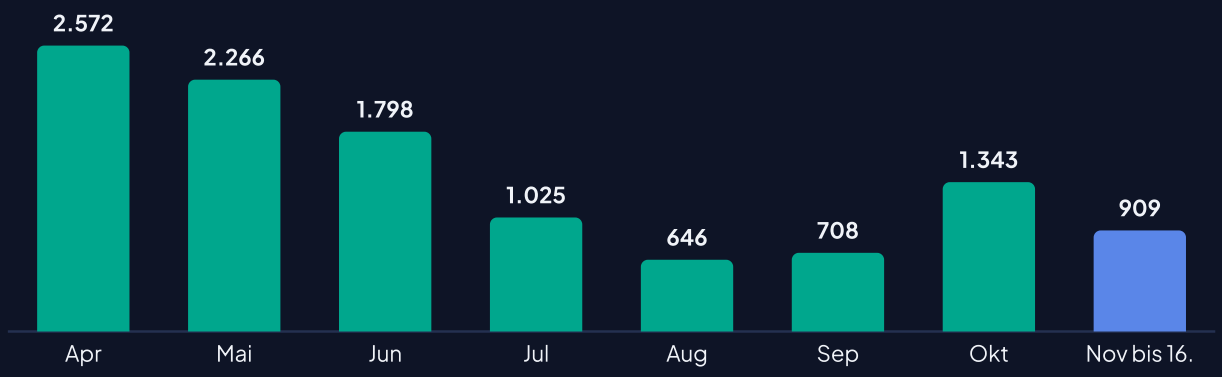


Abbildung 7: Alle Commits einschließlich Zusammenführungen. Im Dezember 2025 gibt es 2 Commits, im Januar bis März 2026 keine: Die Repositories wurden ab April angelegt. Der November reicht bis zum 16.11.

Die Entwicklung begann mit dem Aufbau des Forschungs-Repositories im April 2026. Im Frühjahr entstand die Basis (Sprache, Laufzeit, Modelltraining, Betriebssystem). Ab dem Sommer verlagerte sich die Arbeit auf Website, Untersuchungsberichte und Kundenwerkzeuge, im Oktober und November auf das Managed SOC, Agenten und Auffindbarkeit.

Wie wir die Zahlen einordnen

- **Commits sind kein Maß für Aufwand.** Ein Commit kann eine Zeile ändern oder ein ganzes Modul, daher dienen Commits als Nachweis, dass die Arbeit stattfand.
- **Serien und Massenläufe sind getrennt ausgewiesen.** Im Forschungs-Repository sind 2.492 von 4.689 Commits (53,1 %) serielle Läufe: Bausteine der Standardbibliothek (1.799), automatische Kennzahl-Aktualisierungen (189), Daten-Patches (66) und ein laufendes Beobachtungsprotokoll des Trainings (438). 13 Tage mit je mindestens 100 Commits ergeben dort 48,4 % aller Commits. Ohne diese Serien bleiben 2.197 Commits. Im Betriebssystem-Repository liegen 518 von 527 Commits in acht Tagen (26.05. bis 02.06.2026). In der Website-Historie ergeben die drei stärksten Tage 20 % der Commits.
- **Die Historie beginnt im April 2026.** Frühere Arbeit der Gründergruppe (seit 2020) liegt nicht in diesen Repositories und ist hier nicht mitgezählt.

Forschungs-Repository: 4.689 Commits, davon 53,1 % serielle Läufe



- Bausteine der Standardbibliothek: **1.799**
- Training: Beobachtungsprotokoll: **438**
- Automatische Kennzahl-Updates: **189**
- Daten-Patches: **66**
- Übrige Commits: **2.197**

Abbildung 8: Serielle Läufe (orange und blau) sind getrennt ausgewiesen. Zählweise: Anhang A.

Zusammenarbeit mit KI-Assistenten

Wir arbeiten mit KI-Assistenten (Claude-Modelle von Anthropic, in einzelnen Fällen weitere Werkzeuge) und weisen das in den Commits aus. Auf allen Repositories gilt dieselbe Regel: Wo ein KI-Assistent beteiligt war, steht im Commit die Zeile „Co-Authored-By“ mit dem Namen des Modells. Von Hand gepflegte Commits tragen sie nicht.

BEREICH	COMMITS MIT KI-KENNZEICHNUNG	ANTEIL
Alle Repositories (11.117 inhaltliche Commits)	**5.519**	**49,6 %**
Öffentliche Repositories	**2.786 von 7.311**	**38,1 %**
Private Repositories (Produkte und Werkzeuge)	**2.733 von 3.806**	**71,8 %**
Forschungs-Repository (Ternlang)	**1.217 von 4.689**	**26 %**
Betriebssystem (Rusty Penguin)	**423 von 527**	**80 %**
Website (rfi-irfos.com)	**728 von 1.119**	**65 %**

Commits mit KI-Kennzeichnung (Co-Authored-By)

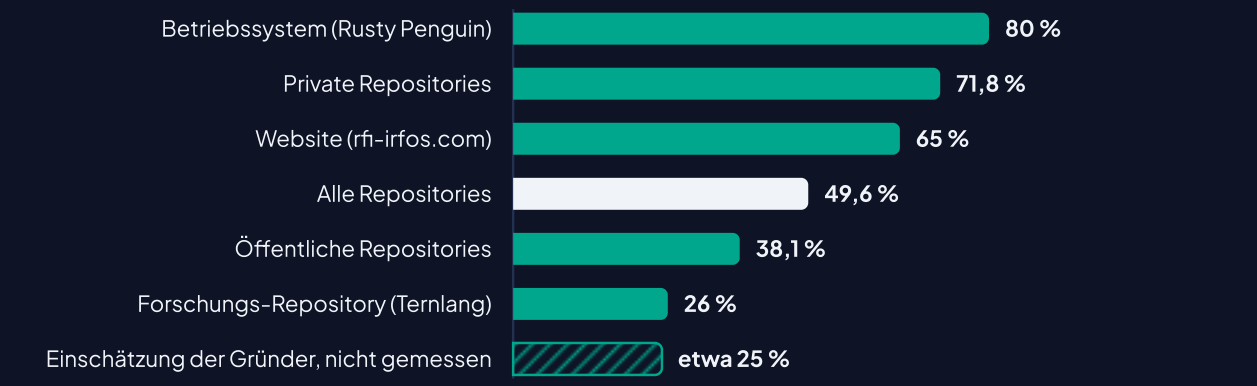


Abbildung 9: Schraffiert ist die Schätzung der Gründer für den Anteil der KI-Arbeit.

Die Zeile zeigt, dass ein Assistent beteiligt war. Sie lässt offen, wie viel von der Änderung von ihm stammt, und steht auch dann im Commit, wenn der Assistent nur die Beschreibung formuliert oder eine von Menschen gebaute Änderung zusammengefasst hat. Nach der Einschätzung der Gründer liegt der Anteil der Arbeit, den

KI-Assistenten tatsächlich erledigen, im Durchschnitt bei etwa einem Viertel, wobei es sich um eine Schätzung ohne Messung handelt. Die Menschen entwerfen die Architektur, entscheiden, was gebaut wird, und prüfen die Ergebnisse. Was das in der Praxis heißt, zeigt das Beispiel der verworfenen Dateien in Kapitel 3.9.1.

Quellen: Git-Historie der Repositories der Organisation rfi-irfos (Hauptzweig, Auswertung am 16.11.2026). Zählweise und Grenzen: Anhang A. Wörtliche Beispiele aus den Commit-Nachrichten: Anhang B.

3.8 Whitebox: nachvollziehbare KI und unsere Sicherheitsrahmen

Warum das jetzt wichtig ist. Die meisten KI-Systeme, die heute im Einsatz sind, sind Blackboxes: Sie liefern eine Antwort, aber nicht den Weg dorthin, und wie sie trainiert wurden, bleibt verborgen. Nachvollziehbarkeit als Eigenschaft des Entwurfs („Explainability by Design“) ist selten. Ein öffentliches Werkzeug, das schon während des Trainings zeigt, was ein Modell gelernt hat, kennen wir von keinem anderen Anbieter. Unsere Arbeit folgt diesem Prinzip seit dem ersten Commit.

Eine KI, die Sicherheitslücken findet, Berichte schreibt oder Entscheidungen vorbereitet, darf keine Blackbox sein. Sonst müssen Betroffene, Betreiber und Aufsichtsbehörden dem Ergebnis glauben, statt es prüfen zu können. Alle unsere Werkzeuge folgen deshalb einem Grundsatz, den wir **Whitebox** nennen: Jeder Schritt, jede Quelle und jede Unsicherheit ist sichtbar und lässt sich nachprüfen. Menschen entscheiden, KI-Agenten bereiten vor. Dieses Kapitel zeigt, aus welchen Bausteinen die Whitebox besteht und wo wir sie im Berichtszeitraum gebaut haben. Das sichtbarste Ergebnis ist Xperience albert., ein vollständig offengelegtes Sprachmodell (Kapitel 3.8.2). Die Belege sind Commits mit Datum. Die Europäische KI-Verordnung verlangt für bestimmte Systeme Transparenz und menschliche Aufsicht, und unsere Bausteine zielen in diese Richtung, ohne dass eine Prüfung auf Konformität stattgefunden hätte.

3.8.1 Die fünf Bausteine der Whitebox

Explainability by Design heißt: Erklärbarkeit liegt von Anfang an im Datenmodell und im Ablauf. Die Logik kennt einen Zustand für „nicht sicher“, und jede Aussage trägt ihren Beleg-Status. Außerdem hinterlässt jede Handlung einen Protokolleintrag, während sich jede Oberfläche selbst erklärt.

BAUSTEIN	FRAGE, DIE ER BEANTWORTET	WAS WIR GEBAUT HABEN (DATUM)
1. Sichtbare Unsicherheit	Weiß das System es, oder rät es?	Die ternäre Logik von Ternlang kennt neben „ja“ und „nein“ einen dritten Zustand „unentschieden, prüfen“ („hold“). Statt eine Antwort zu erzwingen, meldet das System Unsicherheit. Dazu der Entscheidungsprüfer TernAudit für erklärbare Dreiwert-Logik und Konflikte (05.04.), Sicherheitstore mit Veto und mehreren Prüfebenen („safety_gate_3layer“, „veto gate“, 09. und 10.04.), ein Hold-Zustand im Mehrmodell-Baustein ternlang-moe (04.04.).

2. Beweisstatus statt Behauptung	Worauf stützt sich diese Aussage, und ist sie geprüft?	In unserem Forschungs-Cockpit färbt die Whitebox-Ansicht jedes Objekt nach seinem Beleg-Status ein. „Ungeprüft“ erscheint als „noch nicht geprüft“ (23.10.). Evidenz- und Hypothesen-Seiten zeigen zuerst den Status der Aussagen: gestützt, widersprochen, offen (08.11.). Im Untersuchungs-Werkzeug NYX gibt es Beweisklassen mit geschützten Übergängen (06.10.), Hypothesen mit konkurrierenden Erklärungen und Widerlegungs-Kriterien (06.10.) und eine Quellenliste, die unbelegte Quellen standardmäßig abweist („fail-closed“, 05.10.).
3. Verständlich erklärt	Kann jemand ohne Fachwissen nachvollziehen, was er sieht?	Objekte und Gruppen erklärt die Whitebox in einfachem Deutsch (01.11.). Jede Zahl erklärt auf Klick, was sie zählt (06.11.), jede Seite sagt, was sie zeigt, woher die Zahlen stammen und warum (06.–07.11.). Es gilt „ein Wort, eine Bedeutung“, mit Glossar. Rohe Kennungen weichen lesbaren Namen (01. und 07.11.).
4. Nachprüfbare Protokolle	Was ist wann geschehen, und wurde etwas verändert?	Verkettete Protokolle mit Prüfsummen (Hash-Ketten): Lauf-Protokoll für Agenten (04.09.), Untersuchungsprotokoll mit Wiederholung und Prüfbefehl (06.10.), Ereignis- und Herkunftsprotokoll (12.09.), Kontrollprotokoll im Forschungs-Cockpit (25.10.) und eine eigene Kette je Mandant im Fallsystem (02.11.). Eingriffe stehen mit Vorher- und Nachher-Zustand im Protokoll (04.11.).
5. Mensch entscheidet	Wer trägt die Verantwortung für die Handlung?	Befugnis-Matrix mit menschlichem Entscheidungstor (04.09.). Im Fallsystem führen Aktionen über ein Gateway mit Vorschlag, Einmal-Freigabe, Protokoll und einem Tor, das sich nicht umgehen lässt (02.11.). Pull-Requests entstehen nur auf Wunsch je Repository, Freigaben nur durch Administratoren (12.–13.11.). Forschungsläufe zählen Ergebnisse nur mit einer protokollierten Zweitprüfung durch ein anderes Modell (07.11.).

Quelle: Betreffzeilen der Commits in ternary-intelligence-stack (öffentlich) sowie in vier privaten Repositories (Forschungs-Cockpit, Untersuchungs-Werkzeug, Skill-Rahmenwerk, Agenten-Kernel), Auswertung am 16.11.2026. Wörtliche Beispiele: Anhang B.

3.8.2 Xperience albert.: ein vollständig offengelegtes Modell

Die meisten Sprachmodelle sind verschlossen: Man sieht Eingabe und Ausgabe, aber weder den Aufbau noch die Gewichte. Bei unserem Sprachmodell albert. ist es umgekehrt. Unter **albert.rfi-irfos.com** kann jeder Mensch im Browser das ganze Modell als dreidimensionale Ausstellung erkunden. Seitentext, Zahlen und Funktionsweise haben wir am 16.11.2026 auf der Live-Seite geprüft.

WAS DIE SEITE OFFENLEGT	ANGABE AUF DER LIVE-SEITE UND IN DEN COMMITS

Aufbau	Ternäres Zwei-Strom-Modell mit Expertengruppen („mixture of experts“): 46 Schichten je Strom, 2 Ströme, 12 gemeinsame Experten je Schicht, von denen ein Router für jedes Wort 3 auswählt, 7 Austausch-Tore zwischen den Strömen, 331.140.622 Parameter. Jedes Gewicht ist −1, 0 oder +1. Die Aufbau-Datei wird direkt aus dem Kopf des Trainingsstands erzeugt, nicht von Hand gezeichnet (23.10.).
Gewichte	Gewichts-Kacheln aus dem echten Trainingsstand für jede Schicht (23.10.), und die Rechenschritte einer Schicht lesen die 4×4-Ecke der Gewichte (24.10.).
Datenweg	„Follow a token“: Ein Wort wird auf seinem Weg durch das Modell verfolgt. Dazu ein Schritt-für-Schritt-Modus je Schicht (23.10.) und drei gemessene Beispiel-Läufe, aufgezeichnet über Messpunkte im Modell (23.10.).
Training	Das Trainings-Dashboard ist als Ausstellungsstück ausgestellt, später als „Trainingswand“ hinter dem Modell (24. und 25.10.).
Kennzeichnung der Angaben	Jede Angabe trägt ein Etikett: „Struktur“, „gemessen“ oder „illustrativ“ (23.10.). Einschränkungen stehen in einer eigenen Datei „CAVEATS“, die mit dem Hinweistext synchron gehalten wird (24.10.). Der Quelltext und die Modellkarte sind verlinkt, das Modell liegt auf Hugging Face.
Lernen und Zugang	Sieben geführte Rundgänge mit Vorhersage-Momenten: Wer die Antwort vorher rät, versteht sie besser (24.10.). Bedienung auf Deutsch und Englisch, mit Tastatursteuerung (WASD), auf dem Telefon als Seitenleiste, mit Barrierefreiheit, Ersatzseite ohne JavaScript, Lade-Budgets und Rauchtest für jede Veröffentlichung (24.–25.10.).

Quelle: Live-Seite albert.rfi-irfos.com (Abruf 16.11.2026) und Betreffzeilen der Commits im Repository ternary-intelligence-stack vom 23. bis 31.10.2026 (rund 80 Commits zum Explorer). Die Angaben zu Aufbau und Größe stammen vom Projekt.

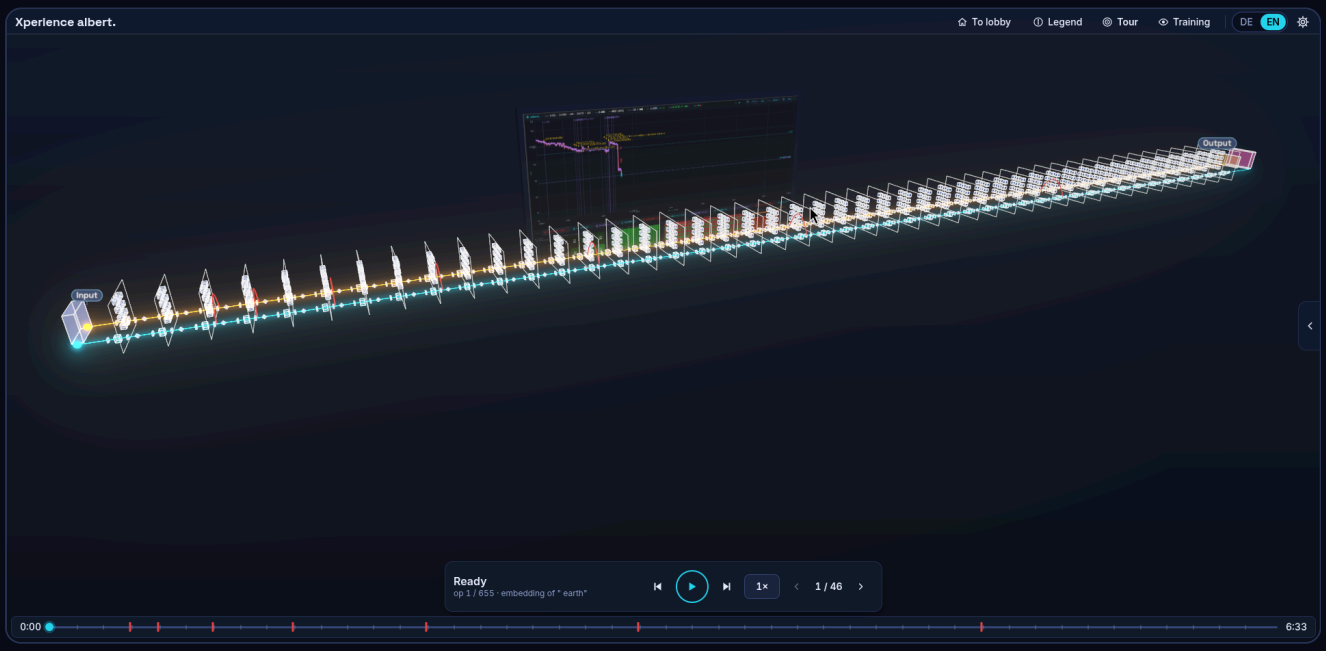


Abbildung 10: Xperience albert. in der Gesamtansicht. Links die Eingabe, rechts die Ausgabe, dazwischen 46 Schichten in zwei Strömen. Im Hintergrund hängt das Trainings-Dashboard. Bildschirmfoto der Live-Seite vom 16.11.2026.

Von oben bis unten sichtbar. Wir verfolgen bei unseren Modellen jedes Wort Schritt für Schritt: Im Beispiel-Lauf der Abbildung sind es 655 Rechenschritte von der Einbettung („embedding“) des ersten Wortes bis zur Ausgabe. Der „Mycelium Inspector“ zeigt zusätzlich den Raum der Einbettungen selbst (16.05.), und Beobachtungs-Läufe sondieren einzelne Einbettungen während des Trainings (Epochen 2510 und 2576, 21.05.). Kein Teil des Modells ist eine verschlossene Kiste. Das gilt vom Training über die Schichten bis zur Ausgabe.

Beispiel: der Einbettungs-Inspector. In unserem internen Forschungs-Arbeitsbereich gibt es zusätzlich einen Inspector für den Einbettungsraum von albert. Der Inspector lädt die Gewichte und Einbettungen des Modells. Wer ein beliebiges Wort eintippt, zum Beispiel „wisdom“, und die Eingabetaste drückt, sieht sofort die 80 nächsten Nachbarwörter, also die Wortzusammenhänge, die das Modell gelernt hat. Dazu gehören ihre Ähnlichkeitswerte (Kosinus-Ähnlichkeit), eine Verteilung der Werte, ein Netz der Verbindungen und eine Ansicht in 2D oder 3D, die man drehen und zoomen kann. Der Raum umfasst 32.000 Wörter mit je 256 Dimensionen und ist auf drei Achsen verkleinert dargestellt (Hauptkomponentenanalyse). So lässt sich ohne Programmierkenntnisse prüfen, was das Modell gelernt hat und ob die Wortnachbarschaften sinnvoll sind.

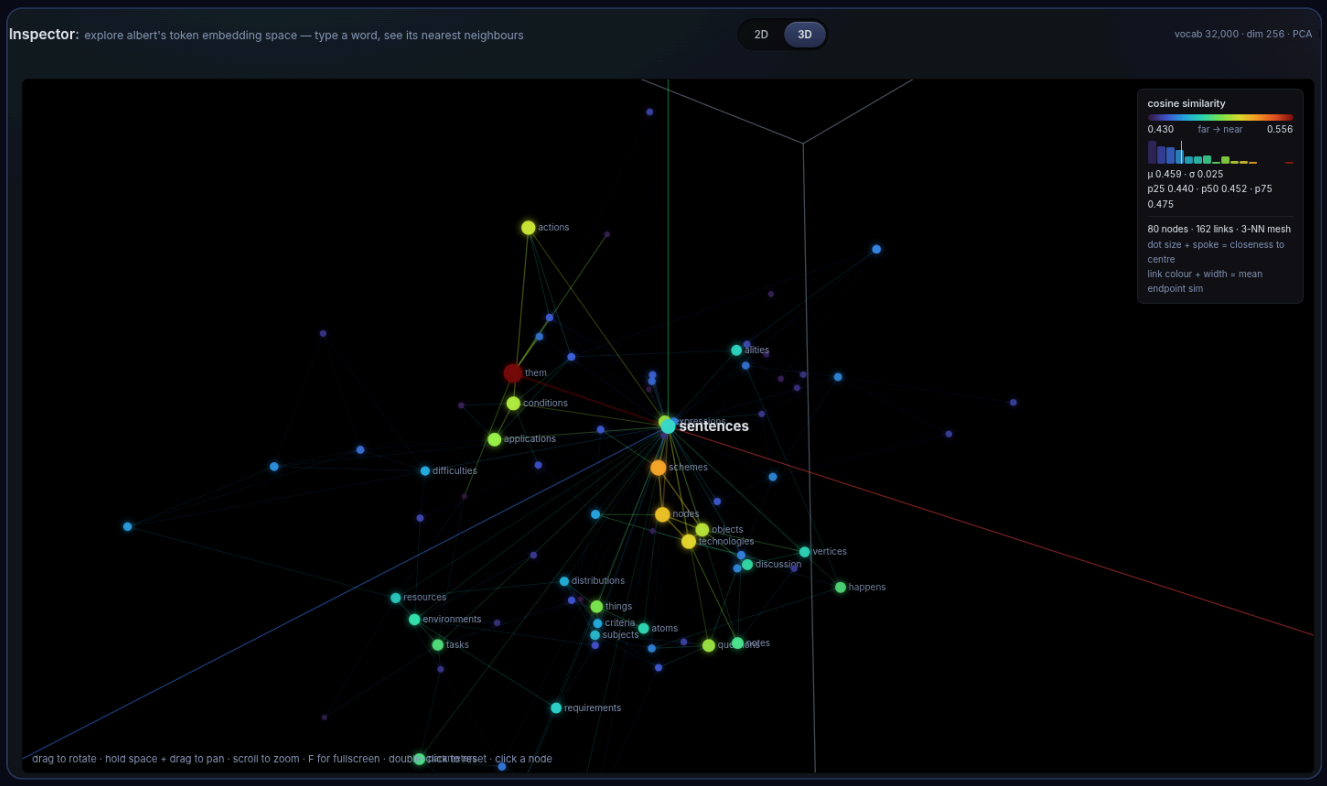


Abbildung 11: Der Einbettungs-Inspector zeigt für das Wort „sentences“ die nächsten Nachbarn im Modell als Netz in 3D. Größe und Farbe der Punkte stehen für die Nähe. Bildschirmfoto vom 16.11.2026 aus dem internen Arbeitsbereich.

Warum das wichtig ist. Wer ein Modell nicht aufmachen kann, muss seinen Herstellern glauben. Mit Xperience albert. kann man eine Sicherheits-, Ethik- oder Lehrfrage am Modell selbst prüfen: Wie viele Experten sprechen mit? Wo tauschen die Ströme Information aus? Was geschieht mit einem einzelnen Wort? Für Schulen, Hochschulen und Fachleute ist das ein Lernwerkzeug. Für Aufsichtsbehörden zeigt es, dass Nachvollziehbarkeit bis in das Modell hinein möglich ist.

3.8.3 Beispiel: die Whitebox-Ansicht im Forschungs-Cockpit

Die sichtbarste Form der Whitebox ist eine Ansicht, in der unser Wissensbestand als Karte aus Objekten, Zusammenhängen und Belegen erscheint. Sie wurde zwischen dem 23.10. und dem 08.11.2026 in rund 40 Commits aufgebaut und mehrfach nach Rückmeldungen umgestaltet:

- **Beleg-Status sichtbar:** Jedes Objekt trägt die Farbe seines Status aus dem Protokoll. Ungeprüftes sieht nicht aus wie Gesichertes.
- **Gruppen nach Zoomstufe:** Beim Herauszoomen fallen Objekte zu Gruppen zusammen, beim Hineinzoomen werden Einzelheiten sichtbar (24.10.). Acht Untersuchungsmodi zeigen denselben Bestand aus verschiedenen Blickwinkeln: Wissen, Denkwege, Beleg und Hypothese, Systeme, Zeitachsen und weitere.
- **Von der Karte zur Aufgabe:** Aus einem gewählten Objekt lässt sich direkt eine Untersuchung starten. Objekt und passender Auftrag sind vorausgefüllt (23.10.).
- **Handy und kleine Fenster:** Die Ansicht funktioniert auch auf dem Telefon (04.11.) und in kleinen Fenstern (06.11.). Eine eigene Layout-Prüfung findet abgeschnittenen Text, zusammengefallene Spalten und zu geringen Kontrast (08.11.).
- **Zählweise:** Eine Zeile am 06.11. korrigiert, dass 20 Monats-Anker nicht als Forschungseinträge mitgezählt werden dürfen. Die Zahlen in der Ansicht sagen „Einträge und Zusammenhänge“ statt einer einzelnen großen Zahl (07.11.).

Dieselbe Idee steckt im Kundenprodukt: Das Portal zeigt Befunde mit Status, Aktivitätsprotokoll und dem Weg von der Aufgabe bis zur Korrektur. Das Ergebnis wird so zu etwas, das ein Kunde selbst prüfen kann.

3.8.4 Sicherheitsrahmen: Schutz in Schichten

Wer Agenten arbeiten lässt, muss ihnen Grenzen setzen. Unsere Sicherheitsrahmen wirken in mehreren Schichten, von den Regeln für KI-Agenten bis zum fertigen Produkt:

SCHICHT	SCHUTZZIEL	UMSETZUNG (DATUM)
Regeln für Agenten	Agenten dürfen nur, was erlaubt ist.	Wächter, der regelwidrige Werkzeug-Aufrufe blockiert, und maschinenlesbare „harte Regeln“ je Skill, Regelprüfung vor jedem Commit (05.08.). Prüfer-Agenten arbeiten mit gesperrten Werkzeugen, ihr Zustandsordner ist gegen Pfad-Tricks geschützt (22.–23.10.). Vor einem Löschen entsteht automatisch ein Schnappschuss (22.10.).
Kern-Prüfung vor jeder Änderung	Keine Änderung ohne Prüfung gegen die Grundsätze.	Ein Agenten-Kern mit Sicherheitshierarchie und Abweichungs-Messung („Drift“, 10.08.) prüft vor dem Push. Der Installer verteilte diese Prüfung auf zehn Repositories der Organisation (12.09.).
Verweigerung und Überwachung	Das System bleibt auch unter Druck bei seinen Grenzen.	Eine unbestechliche Verweigerungsregel und ein Anomalie-Wächter (11.07.). Menschenrechte sind als Grundsatz der Agenten-Doktrin festgeschrieben: „Human rights are not subject to negotiation.“ (18.07.).
Schutz vor Manipulation	Fremder Text darf den Agenten nicht umprogrammieren.	Test mit einer eingeschleusten Anweisung in einem Untersuchungsobjekt gegen ein gehorsames Modell (06.10.). Der Kontext der Modelle ist begrenzt, mit Schwärzung sensibler Teile und Kennzeichnung von Daten (06.10.). Schutz gegen katastrophales Backtracking in Mustern und sichere XML-Verarbeitung (09.10.).

Fail-closed statt fail-open	Im Zweifel wird abgelehnt.	Zahlungs-Webhook lehnt bei Fehlern ab (29.05.), Quellen ohne Beleg werden nicht zugelassen (05.10.), Budget-Sperre bei KI-Programmierassistenten (24.10.), das Forschungs-Cockpit sendet nichts nach außen, solange die europäische Route nicht steht (06.11.).
Prüfung der Prüfung	Tests müssen Fehler wirklich finden.	Mutationstests: Es werden absichtlich Fehler eingebaut, um zu sehen, ob die Tests sie bemerken („none survive“, 23.10. und 03.11.). Rot-Team-Tests für Anmeldung, Skriptinjektion, Datenschutz und Ausfallsicherheit (17.07.). 26 End-zu-End-Prüfungen gegen das laufende System (02.11.).
Vorab festgelegte Forschung	Ergebnisse dürfen schlecht ausfallen.	Prüfprotokolle werden vor der Auswertung eingefroren, jede Regel wird mit der Vorhersage „null“ verglichen, Ergebnisse tragen ein Siegel (04.–06.11.). Ein Ergebnis steht als „nicht bestätigt“ im Protokoll.

Quelle: Betreffzeilen der Commits, Auswertung am 16.11.2026. Beschreibungen sind Aussagen der Projekte (Grenzen: Kapitel 3.8.6).

3.8.5 Messen statt behaupten: Telemetrie rund um Trainings- und Agentenumgebungen

Eine Whitebox braucht Messwerte darüber, was in den Umgebungen passiert, in denen Modelle trainiert und Agenten betrieben werden. Wir haben dafür eigene Messung und Beobachtung gebaut, die der Forschung und der Qualitätssicherung unserer eigenen Systeme dient. Nutzerdaten Dritter werden dafür nicht gesammelt; im Gegenteil, eingebaute Telemetrie ohne klaren Zweck haben wir am 09.04. aus dem offenen Forschungs-Repository wieder entfernt.

UMGEBUNG	WAS GEMESSEN WIRD	UMSETZUNG (DATUM)
Modelltraining (Sprachmodell albert.)	Verlauf des Trainings, Zustand der Rechner, Herkunft der Daten.	Training auf Cloud-GPU mit Überwachung, Modellkarte und dokumentierter Datenherkunft (14.05.). Beobachtungsprotokoll des Trainings mit 438 Einträgen, darunter zurückgezogene Beobachtungen nach Fehlern (27.05.). Beim verteilten Training steht der Verlustwert jedes Beitrags im Protokoll, von 10,21 am 16.05. auf 5,55 am 30.05. Eine Trainings-Anzeige für Rechner ohne GPU erkennt veraltete Meldungen („stale thresholds“, 16.05.). Ein „Mycelium Inspector“ macht den Raum der Wort-Darstellungen sichtbar (16.05.), Reiter für Topologie, Datenfluss, Wörter und Prüfung zeigen das Modell im Betrieb (19.05.). Das Dashboard ist ab 24.10. in Xperience albert. öffentlich ausgestellt (Kapitel 3.8.2).

Agenten-Umgebung	Was jeder Agent tut, mit welchem Modell, wie schnell.	Telemetrie pro Agent mit frei wählbarem Sprachmodell (07.08.), Beobachtungs-Oberfläche mit Trichter vom Auftrag bis zum Ergebnis (07.08.), Ereignis- und Herkunftsprotokoll mit Antwortzeiten von Mensch und Modell (12.09.).
Zustand der Agenten	Driften Agenten vom Auftrag ab? Ist ihr Kontext noch in Ordnung?	Kern mit Messung von Abweichung und Kohärenz („Drift“, „CEI“) und Entscheidungsprotokoll (10.08.). Gesundheits-Anzeige und Drift-Wächter für Agenten, Kontext-Integritätswert mit Wiederherstellungsmodus, Prüfpunkten und Herkunft (05.09.), Umgebungs-Fingerabdruck (06.09.), Systemkarte mit allen Akteuren (06.09.).
Emergenz-Messung	Entsteht wirklich etwas Neues, oder sieht es nur so aus?	Beobachtungs-Instrument mit zehn Modulen (09.07.). Im Code erzwungenes Tor „gemessene Emergenz“ (11.07.). Protokoll, das Ausgabe, Emergenz und Halluzination unterscheidet (19.07.). Je Lauf erfasste Kohärenz-Werte und Resonanz im Schwarm-Reiter des Cockpits (01.11.).
Wiederherstellung	Lässt sich ein Stand wiederherstellen und nachvollziehen?	Prüfpunkte und Ereignis-Wiederholung für einen Agenten (26.08.), versionierte Prüfpunkte je Zone im Untersuchungs-Werkzeug (22.10.), Wiederherstellung mit Prüfung der Prüfsumme, die beschädigte Stände ablehnt (10.08.).

Quelle: Betreffzeilen der Commits in ternary-intelligence-stack, albert-spores (beide öffentlich) und privaten Repositories (Agenten, Skill-Rahmenwerk, Forschungslabor, Forschungs-Cockpit, Untersuchungs-Werkzeug), Auswertung am 16.11.2026.

Training live beobachten, bei Auffälligkeiten pausieren, danach einfach weitermachen. Das Training von albert. läuft auf gemieteten Cloud-GPUs der Plattform Modal, und wir sehen dem Modell dabei in Echtzeit ins Innere und greifen ein, sobald etwas auffällig aussieht. Im Einzelnen gilt:

- **Live-Beobachtung:** Ein Dashboard mit Echtzeit-Verbindung (WebSocket) zeigt Verlustkurve, Ereignisse und Zustand des Trainings (03.05.). Ein Wächter für den Datenstrom und die Erkennung veralteter Anzeigefelder sorgen dafür, dass kein Stillstand unbemerkt bleibt (13.05.). Ein Ereignis-Streifen meldet Anomalien (07. und 11.05.). Der Einbettungs-Inspector (siehe oben) lädt dabei die jeweils frischen Gewichte, zuletzt aus Epoche 6487 (19.06.).
- **Pausieren bei Auffälligkeit:** Wenn das Training stockt, stoppt ein Wächter nach 180 Sekunden Stillstand den Lauf sauber („clean stop“), und die Gewichte bleiben unversehrt (27. und 29.05.). Auch von Hand lässt sich das Training anhalten, zum Beispiel für einen Eingriff am Modell, und danach wieder starten (14.05., 28. und 30.05.).
- **Fortsetzen statt neu beginnen:** Nach einem Stopp setzt das Training am letzten Stand fort. Am 27.05. stand das Training um 18:26 Uhr (UTC) still und lief um 18:40 Uhr wieder. Das 26-schichtige Modell überstand die Wiederaufnahme ohne Speicherfehler (28.05.), und der Verlust blieb nach dem Wiederanlauf auf gleichem Niveau (29.05.).

- **Geringe Ausfallkosten:** Die Plattform rechnet sekundengenau ab. Ein gestoppter Lauf kostet deshalb nur die Sekunden, in denen er lief, und es muss nicht das ganze Training noch einmal von vorn bezahlt werden.
- **Auch Fehlalarme sind protokolliert:** Am 27.05. meldete unser Beobachtungsprotokoll einen Trainingsstopp, der keiner war. Nur die Benachrichtigung war ausgeblieben. Der Eintrag wurde zurückgezogen und als Korrektur festgehalten.

Quelle: Betreffzeilen der Commits in ternary-intelligence-stack vom 03.05. bis 19.06.2026, Beobachtungsprotokoll des Trainings und Eigenangabe zur sekundengenauen Abrechnung der Cloud-Plattform.

Der Nutzen liegt in der Verbindung der drei Dinge: Das Training liefert Messwerte, die Agenten-Umgebung zeigt, was die Modelle in der Praxis tun, und die Whitebox macht beides für Menschen lesbar. So lässt sich bei Auffälligkeiten rückverfolgen, ob Daten, Training, Agent oder Aufgabe die Ursache sind.

3.8.6 Was uns die Praxis gelehrt hat

- **Ein neuer Schutz löst zuerst den eigenen Ausfall aus.** Bei jedem neuen Schutzmechanismus war der erste Einsatz das Blockieren eines legitimen eigenen Vorgangs. Deshalb gilt seit dem Sommer: Ein Schutz bekommt einen erlaubten Gegenweg, einen Test, der ihn absichtlich auslöst, und eine möglichst frühe Prüfstelle (Entwurf, Test, Freigabe, Betrieb). Er blockiert nur den betroffenen Vorgang und nicht das ganze System.
- **Agenten brauchen Aufsicht auch über sich selbst.** Aus den Korrekturen (verworfene maschinell erzeugte Dateien, zurückgenommene Messangaben) folgte die Regel, dass Forschungsergebnisse nur mit Zweitprüfung durch ein anderes Modell zählen (07.11.).
- **Verständlichkeit ist Teil der Sicherheit.** Ein Befund, den niemand versteht, kann niemand prüfen. Daher die Arbeit an einfacher Sprache und erklärenden Zahlen in allen Oberflächen.

Grenzen. Die Whitebox macht Fehler sichtbar, sie verhindert sie nicht. Die genannten Rahmen sind von uns entworfen und getestet, nicht von Dritten zertifiziert. Wir planen für 2027 eine unabhängige Begutachtung der wichtigsten Rahmen (Kapitel 4.1).

3.9 Fallstudien aus dem Berichtszeitraum

Die folgenden Fallstudien zeigen an fünf Beispielen, wie aus einer Idee ein Ergebnis wurde: was gebaut wurde, in welcher Reihenfolge, was schiefging und wie wir es korrigiert haben. Datum und Beschreibungen stammen aus den Commit-Nachrichten. Aussagen der Projekte über sich selbst sind als solche gekennzeichnet.

3.9.1 Ternlang und das Rahmenwerk für sparsame KI (Repository ternary-intelligence-stack)

Leitfrage: Kann KI mit viel weniger Speicher und Rechenaufwand arbeiten, wenn sie statt mit zwei mit drei Zuständen rechnet (−1, 0, +1)? Um das zu prüfen, brauchten wir eine eigene Programmiersprache, eine Laufzeit dafür und ein Modell, an dem man messen kann. Das Repository ist öffentlich (LGPL-3.0). Es enthält 4.689 inhaltliche Commits an 103 Tagen zwischen dem 01.04. und dem 13.11.2026.

NR.	ZEITRAUM	COMMITS	WAS ENTSTAND (BELEGT MIT DATUM)
1	01.–07.04.	**255**	Grammatik und Sprachreferenz (01.04.). Aufrufe in der virtuellen Maschine, Tensor-Befehle, Akteur-Modell und erster Server für KI-Werkzeuge (02.04.). Editor-Unterstützung, Paketverwaltung und HTTP-Server, Veröffentlichung auf crates.io (03.04.). Werkzeuge zum Übertragen und Prüfen von Entscheidungslogik (05.04.).

2	08.–10.04.	**468**	Erste Versionsmarken mit Codenamen (08.04.). Entfernung eingeschleuster Telemetrie (09.04.). Bereinigung: 3.000 maschinell erzeugte Dateien der Standardbibliothek werden nach Prüfung verworfen und durch echte Inhalte ersetzt, zudem wird ein erfundener Build-Ablauf durch Build-Schritte abgelöst (10.04.).
3	11.–17.04.	**1.147**	Fehlerlisten mit Regressionstests (15.04., 21 Einzel-Commits), Sammelkorrektur von 14 Fehlern (16.04.), Bereinigung weiterer maschinell erzeugter Reste (16. und 17.04.). Erste stabile Version v1.0.0 mit 30 Werkzeugen für KI-Assistenten (17.04.).
4	18.–30.04.	**697**	Die virtuelle Maschine läuft im Webbrowser über WebAssembly (18.04.). Start von TernStudio, einem visuellen Baukasten für Agenten. Kommandozeile für das Sprachmodell albert. (25.04.). Entfernen eines fest eingetragenen Entwickler-Tokens (28.04.).
5	01.–06.05.	**622**	Erster ternärer Ausführungspfad im Modell. Rücknahme einer eigenen Leistungsangabe auf eine theoretische Obergrenze (02.05.). Veröffentlichung des gesamten Stacks auf crates.io, 34 Werkzeuge für KI-Assistenten (06.05.).
6	07.–20.05.	**623**	Sicherheitswelle: Schutz gegen Pfad-Tricks, Ratenbegrenzung, Prüfsummen für eingebundene Skripte, Härtung der virtuellen Maschine und des Schlüsselspeichers (11.–16.05.). Training auf Cloud-GPU mit Überwachung, Modellkarte und Datenherkunft dokumentiert (14.05.).
7	21.05.–15.06.	**702**	Das Modell wächst Schicht für Schicht von 17 auf 21 Schichten, danach auf 26 mit zwei Datenströmen (21.–27.05.). Version v3.0.0 (28.05.). Stripe-Webhook wird „fail-closed“: Bei einem Fehler wird abgelehnt statt durchgelassen (29.05.). Version v3.1.0 (15.06.).
8	16.06.–20.09.	**98**	Betrieb und Pflege bei niedriger Commit-Dichte: security.txt für Sicherheitsmeldungen (20.06.), ternärer episodischer Speicher für KI-Assistenten (25.06.), Umstellung auf eine stärkere GPU und weiteres Modellwachstum (16.–20.09.).
9	12.10.–13.11.	**77**	Interaktive 3D-Ansicht des Modells „Xperience albert.“ im Browser, live unter albert.rfi-irfos.com (24.10.), mit sieben geführten Rundgängen, Tests und Telefonansicht. Version v3.2.0 (25.10.). Aktualisierung einer Abhängigkeit nach Warnung des eigenen Sicherheitsportals (13.11.).

Ternary Intelligence Stack: Commits je Phase

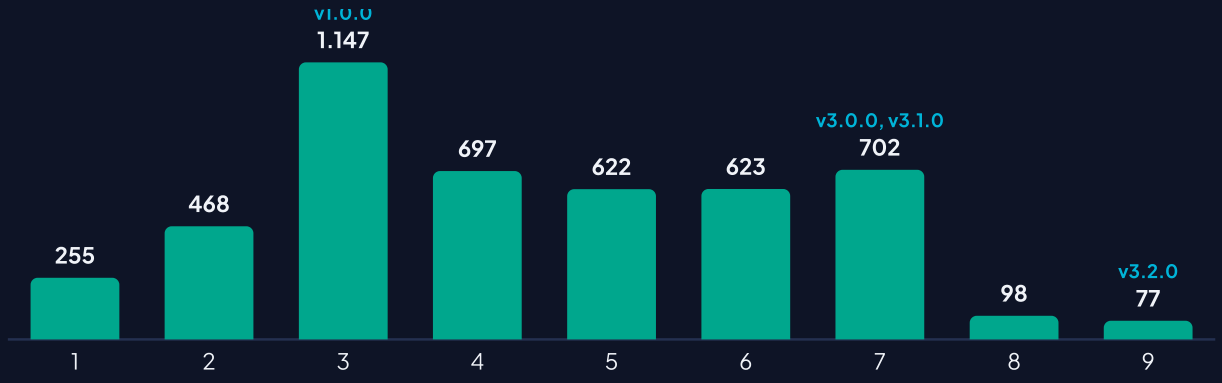


Abbildung 12: Phasen nach der Tabelle: 1 = 01.–07.04., 2 = 08.–10.04., 3 = 11.–17.04., 4 = 18.–30.04., 5 = 01.–06.05., 6 = 07.–20.05., 7 = 21.05.–15.06., 8 = 16.06.–20.09., 9 = 12.10.–13.11. Marken sind Versionen.

Quelle: Betreffzeilen der Commits und Tags im Repository, Auswertung am 16.11.2026. Phasen sind unsere Gliederung nach Datum und Inhalt. Aussagen zu Normen oder Patenten in einzelnen Commit-Nachrichten sind Eigenangaben des Projekts und hier nicht als geprüfte Tatsachen übernommen.

Was die Chronik zeigt. Das Projekt wuchs schnell und räumte ebenso offen auf. Mehrere Commits sind Rückbau und Korrektur: die 3.000 verworfenen Dateien (10.04.), die Rücknahme einer 122-fachen Geschwindigkeitsangabe auf eine theoretische Obergrenze (02.05. und 03.05.), die Korrektur von Parameterangaben in Roadmap und Whitepaper (22.05.) und die Rücknahme zweier Trainingsbeobachtungen mit dem Vermerk „CORRECTION“ (27.05.). Aus den Messungen, die den Prüfungen standhielten, entstanden die Messwerte in Kapitel 3.2: 4× weniger Speicher, 1,63× Durchsatz.

3.9.2 Rusty Penguin: ein Betriebssystem von Grund auf (Repository rusty-penguin)

Ausgangsüberlegung: Wenn ternäre Logik in jeder Schicht eines Rechners gelten soll, braucht man ein Betriebssystem, das sie von Anfang an kennt. Rusty Penguin ist ein solches Betriebssystem für x86_64-Rechner in reinem Rust: eigener Kernel, Treiber, Fenstersystem und Anwendungen, ohne Linux-Kernel darunter (MIT-Lizenz, 527 inhaltliche Commits zwischen 26.05. und 14.08.2026). Der Auslöser war ein Konferenzvortrag, in dem jemand scherzhaft vorhersagte, irgendwann starte jemand ein eigenes Betriebssystem in Rust. Das Projekt hat das Dokument dazu selbst abgelegt (ORIGIN).

NR.	ZEITRAUM	WAS ENTSTAND (BELEGT MIT DATUM)
1	26.–27.05.	Kernel startet über den Bootloader GRUB und schreibt auf den Bildschirm. Es folgen Unterbrechungen, Tastatur, Speicherverwaltung, Wechsel in den geschützten Benutzerbereich mit Systemaufrufen, Lader für Programmdateien und eine eigene Shell. Der Tag endet mit einer startfähigen ISO-Datei, die in der virtuellen Maschine QEMU bootet, und der Einbindung der KI-Bausteine in den Kernel (26.05.). Grafischer Desktop mit Maus und Taskleiste.
2	28.05.	144 Commits an einem Tag: Taschenrechner, Texteditor, Dateimanager, Prozessmonitor, Paketverwaltung mit Prüfsumme und digital signierter Paketliste, Installation auf Festplatte.

3	29.–30.05.	Netzwerk in sechs Bausteinen (Ethernet, IPv4, UDP mit DHCP, TCP mit HTTP-Abruf, Namensauflösung), USB- und Audiotreiber, ein selbst geschriebener TLS-1.3-Client für HTTPS (30.05.), SATA-Plattentreiber, eigenes Dateisystem, WLAN, Mehrbenutzer-Anmeldung.
4	31.05.–01.06.	Umzug des Kernels in den oberen Adressbereich, als „VERIFIED“ geführt. Freigabe von Speicher statt nur Vergabe, Prüfung von Zertifikatsketten, Isolierung hängender Programme, Grafiktreiber, ACPI-Abschaltung und Neustart.
5	02.06.	145 Commits: Ein bestehendes Spielprogramm läuft in einem Desktop-Fenster. Dazu eine Schicht für Windows-Programme, ein Treiber-Rahmen, ein eigener Webbrowser mit eigenem CSS- und JavaScript-Interpreter. Anlegen des Entwicklungsprotokolls „SESSION_LOG.md — honest per-session build record“.
6	05.06.–14.08.	Pflege der Dokumentation und Mitwirkendenliste. Letzter inhaltlicher Eintrag ist ein Entwurf mit Umsetzungsplan für eine ternäre Graphen-Anwendung (14.08.).

Quelle: Betreffzeilen der Commits und Tags im Repository, Auswertung am 16.11.2026.

Was die Chronik zeigt. Der Großteil der Arbeit liegt in acht Tagen (518 von 527 Commits), mit Tagen von 144 und 145 Commits. Das Projekt unterscheidet Prüfstufen selbst: Zwei Commits sind als „[UNVERIFIED]“ markiert, 15 als „VERIFIED“. Viele Prüfungen laufen in der virtuellen Maschine QEMU, und ob alle Funktionen auf echten Geräten laufen, belegen die Commit-Nachrichten nicht; hierzu schreibt das Projekt selbst „honest status update“ und „honest cert caveat“ (30.05.). Aussagen wie „Chrome works“ sind Aussagen des Projekts. Die Mitwirkendenliste weist die Zusammenarbeit mit KI-Assistenten offen aus.

3.9.3 Website und Untersuchungsberichte (Repository rfi-irfos-web)

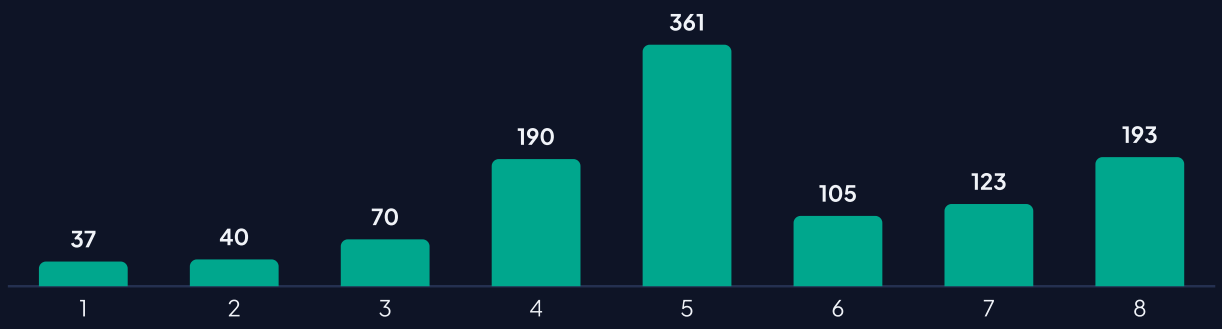
Leitfrage: Wie machen wir Forschung so zugänglich, dass Betroffene, Betreiber und Aufsichtsbehörden sie prüfen können? Die Website rfi-irfos.com zeigt die Untersuchungen mit Status, Frist und Rechtsgrundlage und stellt die Berichte als PDF bereit. Sie hat 1.119 inhaltliche Commits an 105 aktiven Tagen zwischen 04.06. und 16.11.2026 (öffentliches Repository).

NR.	ZEITRAUM	COMMITS	WAS ENTSTAND (BELEGT MIT DATUM)
1	04.–21.06.	**37**	Zuerst ein Baukasten für Websites mit Bearbeitungsbereich, ein Besucherzähler und ein Mitgliederbereich. Zugangsdaten werden aus dem ausgelieferten Webpaket entfernt (16.06.).
2	22.–25.06.	**40**	Start der Institutsseite mit eigener Domain und erzwungener HTTPS-Weiterleitung. Rechtstexte, Kontaktformular, Veröffentlichungsliste und Zeitleiste mit 23 Meilensteinen. Erstes Ledger (öffentliches Verzeichnis der Untersuchungen) mit 50 Zielen und Statusanzeige (22.06.). Begrenzung der Anfragen je Absender (25.06.).
3	26.06.–04.07.	**70**	Ledger mit Filtern, Fristzähler je Zeile und 113 Einträgen (26.06.). Versanddaten werden aus dem Postfach nachgezogen. Aufsichtsbehörden stehen bei jeder Meldung in Kopie (30.06.). Öffentlicher Meldeweg für Hinweise (02.07.).
4	05.–31.07.	**190**	Methodik-Abschnitt (07.07.). Rechtsnormen je Untersuchung in 24 Stapeln (11.–18.07.). Zweites Verzeichnis zu Menschenrechten mit 66 Einträgen, davon 13 in der EU und 53 weltweit (24. und 27.07.).

5	August	**361**	Zweisprachigkeit Englisch und Deutsch (03.08.). Sitemap und strukturierte Daten (04.08.), vorab erzeugte Seiten, damit Suchmaschinen den Inhalt sehen (09.08.). Korrektur der App-Zahl von „315/649“ (ungeprüft) auf „311“ (geprüft) (12.08.). Kennzahlen werden aus dem Berichtsverzeichnis berechnet statt von Hand gepflegt (21.08.).
6	01.09.–12.10.	**105**	Veröffentlichung von Berichten nach Fristende, einzeln (ab 23.08. und 01.09.) und als große Stapel (18. und 19.09.: unter anderem 42 und 29 Untersuchungsziele). Seite „The Squad“ und „World Model“. Dreifache Korrektur eines Berichts nach erneutem Lesen (20.09.).
7	13.–30.10.	**123**	Prüfangebot (heute „Domain Live Testing“) mit Eingabeformular, Freigabe-Warteschlange und Begrenzung auf 3 Anfragen je Absender und 24 Stunden (14.10.). Schriften selbst gehostet, Sicherheitskopfzeilen, Adresse für Offenlegungen und security.txt (21. und 24.10.). Schließen aller 9 Dependabot-Warnungen (26.10.).
8	31.10.–16.11.	**193**	Prüfrunde am 31.10. (119 Commits): Barrierefreiheit (Sprung zum Inhalt, sichtbarer Tastaturfokus, bessere Farbkontraste), Suchmaschinen-Daten, deutsche Gegenstücke der Seiten, Datenschutzerklärung. Atom-Feed der 50 neuesten Berichte. Anfragen aus Quick Check und Kontaktformular landen als Interessent im Kundenwerkzeug (14.11.).

Quelle: Betreffzeilen der Commits im Repository, Auswertung am 16.11.2026.

Website und Berichte: Commits je Phase



50
Einträge am 22.06.

113
Einträge am 26.06.

387
Einträge heute, 249 Berichte als PDF

Abbildung 13: Phasen nach der Tabelle: 1 = 04.–21.06. bis 8 = 31.10.–16.11. Unten wächst das öffentliche Verzeichnis der Untersuchungen in Wellen.

Wie die Untersuchungsberichte gepflegt werden. Das Verzeichnis wuchs in Wellen von 50 Einträgen (22.06.) über 113 (26.06.) bis zum heutigen Stand von 387 Einträgen und 249 veröffentlichten Berichten. Die Commit-Nachrichten dokumentieren das Vorgehen:

- **Fristen und Kopie an Behörden.** Betreiber erhalten den Bericht vor der Veröffentlichung, Aufsichtsbehörden stehen bei jeder Meldung in Kopie (seit 30.06.), und die Berichte erscheinen am Tag

des Fristendes. Zudem werden Versanddaten mit der tatsächlich zugestellten Korrespondenz abgeglichen (26.06., 18.08., 31.08., 19.09.).

- **Reaktionen werden mit Status geführt** (zum Beispiel bestätigt, inhaltlich geantwortet, eskaliert, gelöst, keine Antwort), mit Zähler für die Tage ohne Antwort (05.08.). Die Statusfarben sind bewusst neutral gehalten: „no emotional undertone by status“ (17.07.).
- **Rechtsgrundlage je Untersuchung.** Ab 11.07. nennt das Verzeichnis die betroffenen Rechtsnormen, wofür an zwei Tagen (17. und 18.07.) 24 Stapel entstanden.
- **Öffentliche Spielregeln.** Methodik-Karten (07.07.), Hinweis auf den Verzicht auf Stör- und Lasttests, Bewertungsmethodik im Fuß des Verzeichnisses (20.09.), eigene Adresse für Offenlegungen (21.10.).
- **Prüfschranke gegen Datumsfehler.** „Report dates never precede publication“: Ein Berichtsdatum darf nie vor der Veröffentlichung liegen (31.10.).

Die Historie enthält auch den Aufwand, den zu viel Gestaltung verursacht: 168 Commits betreffen Animationen, Bewegung und Optik, und 16 enthalten „revert“, also Ausprobieren und Zurücknehmen. Wir zählen das offen mit, weil es zur Wahrheit der Arbeit gehört.

3.9.4 Weitere offene Werkzeuge und Kleinprojekte

Neben den großen Projekten stehen kleine, abgeschlossene Werkzeuge öffentlich zur Verfügung. Jedes beantwortet eine konkrete Frage:

REPOSITORY	ZWECK	ZEITRAUM, COMMITS	LIZENZ
pass-the-bot	Zeigt, wie ein Bewerbermanagement-System einen Lebenslauf liest: Stellenanzeige einfügen, Lebenslauf hochladen, transparenten Stichwort-Abgleich erhalten, nach Angabe des Projekts ohne Konto und ohne Speicherung. Entstanden mit Entwurf, Umsetzungsplan und Abschluss-Review. Aus einem Absturz durch Speichermangel (29.09.) wurde die Konsequenz gezogen, den Dienst größer auszustatten. Am 10.11. folgten Schutz vor Überlastung und vor Umlenkungs-Tricks beim Abruf von Links.	14.09.–10.11.2026, 125; Tags v0.2.0 bis v0.3.1	MIT
albert-spores	Sammelstelle, über die Beitragende das Modell albert. auf eigenen Rechnern mittrainieren und Zwischenstände („Spores“) einreichen. Der Verlustwert (Fehlermaß) fällt in den Einträgen von 10,21 (16.05.) auf 5,55 (30.05.).	15.05.–24.07.2026, 66	keine
gut-check	Prüftor für KI-Agenten mit zwei Schleifen: Ein schneller Klassifikator markiert Unsicherheit und reicht sie an das große Sprachmodell weiter. Eine Messung an Aufgaben mit bekannter Lösung ergab laut Commit-Nachricht „found a real failure“ (22.09.).	22.–23.09.2026, 32	Apache-2.0
invisible-layer	Webseite mit 44 Sensor-Experimenten, die zeigen, was ein Telefon alles messen kann (Seismik, Raumakustik, Kompass, Barometer und weitere). Zweisprachig, installierbar, mit Barrierefreiheitsdurchgang.	23.05.–06.10.2026, 85	keine

claude-cache-idiocracy	Werkzeug, das beim Arbeiten mit KI-Programmierassistenten den Verbrauch senkt: Es vermeidet doppelte Lesezugriffe, schützt Zwischenspeicher, verdichtet lange Sitzungen, begrenzt gleichzeitige Unter-Agenten und sperrt bei Fehlern („fail-closed“).	24.–26.10.2026, 42	MIT
call-laura, lauras-port-proxy, laura, veo-framework, lauras-agents-public	Open-Source-Arbeiten aus der Forschung von Laura Serna Gaviria am Emergent Interaction Lab, zwei davon mit eigenem Paper (Kapitel 3.11): ein deterministischer Prüfserver für KI-Assistenten, eine installierbare Prüfung offener Netzwerk-Ports, ein Köder-Datensatz gegen Datendiebstahl über Nähe und ein Verfahren, mit dem ein Sprachmodell Ausgabe, Emergenz und Halluzination auseinanderhält. Für den Prüfserver wurde eine Sicherheitslücke bei den Zugriffsregeln (CORS) geschlossen (16.07.).	12.07.–19.08.2026, zusammen 53	MIT, teils nicht eindeutig
agent-albert-cli, bifp, ai-security, ginie	Kommandozeilen-Programm für mehrere Sprachmodelle, Protokollentwurf für wechselseitige Rückmeldungen, Sammlung von 20 Werkzeugen zur Sicherheitsprüfung von KI-Systemen und ein Desktop-Begleiter, der offline von einem USB-Stick läuft.	22.06.–06.10.2026, zusammen 47	MIT bzw. keine
foodchain-analysis	Agentenbasiertes Simulationsmodell zum globalen Ernährungssystem: Es rechnet durch, wie sich das Kalorienangebot von der Ernte bis zum Teller verringert (Verluste nach der Ernte, Verarbeitung, Zugangshürden). Das Modell kommt zu dem Ergebnis, dass die Erde rund das 1,64-Fache des Kalorienbedarfs erzeugt, das wirksame Angebot aber durch diese Verluste auf etwa das 0,94-Fache sinkt. Das ist eine Aussage des Modells und keine unabhängig geprüfte Messung. Mit Dashboard im Browser. Forschungsergebnis zum Thema Umwelt und Ernährung (SDG 2 und 12).	17.06.–15.08.2026, 14 (auch Rück- und Neuveröffentlichung)	keine Angabe
nikoletta-tutor	Website für ein Sprachcoaching, gebaut mit dem eigenen Baukasten samt Bearbeitungsbereich für die Inhaberin.	2026, 153	keine

Quelle: Betreffzeilen der Commits, Repository-Beschreibungen und Lizenzangaben auf GitHub (Abruf 16.11.2026). Beschreibungen („ohne Konto, ohne Speicherung“) sind Aussagen der Projekte.

3.9.5 Wissensvermittlung: Vortragsreihe und Workshops

Das programmierbare Brot. Wir haben eine Vortragsreihe zu der Frage entwickelt: Wenn Identität, Körper und Geld zu programmierbaren Infrastrukturen werden, wer schreibt dann den Code und wer bestimmt die Regeln? Der Vortrags-Codex 1.0 umfasst 8 Module mit 14 Vorträgen zu digitaler Identität (europäische Wallet), Zentralbankgeld und programmierbarem Geld, KI-gestützten automatisierten Entscheidungen, dem Internet der Körper und Resilienz. Er ist so gebaut, dass jedes Modul einzeln für Universitäten, Vereine oder Seminare einsetzbar ist, vom kurzen Einsteigerformat bis zur mehrtägigen Konferenz. Ein Sponsoring-Paket und ein Flyer sind erstellt.

Ein Symposium in Graz, bei dem die Reihe als Veranstaltung gezeigt wird, planen wir als Rollout für das kommende Jahr (2027). Inhaltlich ist die Reihe fertiggestellt und steht Veranstaltern bereit. Da öffentliche Aufklärung über digitale Infrastruktur eine eigene Finanzierung braucht, klären wir diese, bevor wir einen Termin ansetzen.

Austausch mit Sindbad. Mit der Organisation Sindbad (Mentoring für Jugendliche) stehen wir im Austausch. Für deren Mentorinnen und Mentoren haben wir einen etwa zweistündigen Vortrag zu KI gehalten. Bei einem Abendtreff gab es außerdem einen Workshop in informellem Rahmen. Beides waren Einzeltermine und keine Veranstaltungsreihe.

Workshop-Formate. Für solche Einsätze haben wir Formate entwickelt, die KI begreifbar machen. „AI zum Anfassen“ dauert 90 Minuten und übersetzt einen Alltagsvorgang in die digitale Welt: Die Teilnehmenden legen gemeinsam Gurken ein und erleben dabei, wie Zutaten zu Datenobjekten, ein Rezept zum Algorithmus und das Einmachglas zur Datenbank werden. Die „KI-Degustation“ behandelt KI wie eine Weinprobe in vier „Weingängen“ und drei „Brotphasen“ der gemeinsamen Reflexion. Zu den Formaten gehören Regiebuch, Ablaufpläne und ein Prompt-Pack für die Teilnehmenden.

Social Media. Bis Oktober 2026 haben wir fast nur auf LinkedIn veröffentlicht, doch seit dem 30.10.2026 planen und veröffentlichen wir mit unserem eigenen Werkzeug FluxPost auf fünf Kanälen. Bis zum 16.11.2026 sind das 65 Beiträge (aus dem Export von FluxPost, Testbeiträge nicht mitgezählt):

KANAL	BEITRÄGE SEIT 30.10.	AUFRUFE ODER REICHWEITE	STAND DER FOLLOWER
LinkedIn	**23**	1.069 Impressionen, 623 erreichte Personen mit diesen Beiträgen; 74 Klicks auf Links	800 (798 am 08.11.)
YouTube	**8**	15.329 Aufrufe, 194 „Gefällt mir“. Das erfolgreichste Video hat 6.130 Aufrufe.	105 (7 am 08.11.)
Instagram	**18**	111 Aufrufe, 99 erreichte Konten	1
Facebook	**13**	12 Aufrufe	0
TikTok	**3**	noch keine Messwerte (Veröffentlichung im Prüfverfahren der Plattform)	–

65 Beiträge auf fünf Kanälen seit 30.10.2026

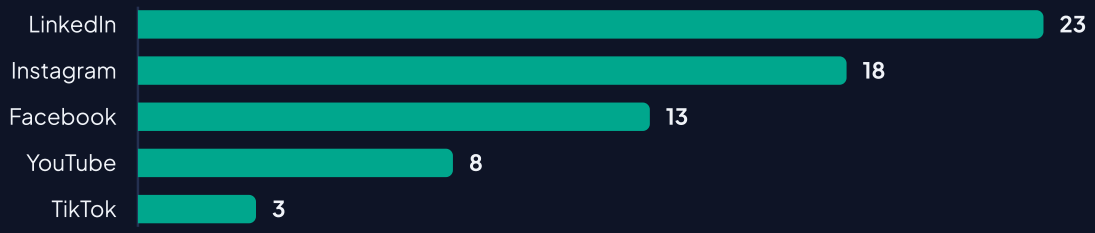


Abbildung 14: Quelle: FluxPost-Export vom 16.11.2026, Testbeiträge nicht mitgezählt. Aufrufe und Reichweite zählt jede Plattform anders, deshalb stehen sie in der Tabelle und nicht addiert in dieser Abbildung.

Quelle: FluxPost-Export vom 16.11.2026 (Zeitraum 30.10. bis 16.11.2026, Messungen der Plattformen) und Tageswerte der Plattform LinkedIn: Für den längeren Zeitraum 04.08. bis 15.11.2026 meldet LinkedIn 11.317 Impressionen und 6.795 eindeutige Impressionen, 313 Klicks, auch für Beiträge ohne FluxPost. Ein Follower-Verlauf liegt erst ab 08.11.2026 vor.

Die Zahlen sind klein, und wir weisen sie auch so aus. Die Kanäle laufen erst seit zwei Wochen, und die Beiträge sind der Beginn einer regelmäßigen fachlichen Kommunikation, die wir 2027 ausbauen und auswerten (Kapitel 4.1). Ältere Beiträge stehen nicht in der Tabelle.

3.10 Alle Repositories im Überblick: vom Fundament bis zur Anwendung

Die Organisation rfi-irfos führt 59 Repositories: 23 eigenständige öffentliche, 18 öffentliche Forks (eines davon archiviert) und 18 private. Dieses Verzeichnis beschreibt jedes von ihnen, in der Reihenfolge, in der die Arbeit aufeinander aufbaut. Oben liegt das Fundament, unten die Anwendungen. Stand aller Angaben: 16.11.2026, Quelle GitHub-Schnittstelle und Repository-Beschreibungen. Selbstaussagen der Projekte sind wie in Kapitel 3.9 einzuordnen.



Abbildung 15: Durchgezogene Kästen sind öffentliche, gestrichelte Kästen sind private Repositories. Forks und das Archiv stehen in Kapitel 3.10.3.

3.10.1 Eigenständige öffentliche Repositories

Fundament: Sprache, Laufzeit und Modell

REPOSITORY	WAS ES IST	LIZENZ LAUT GITHUB, SPRACHE	ZEITRAUM, STERNE
ternary-intelligence-stack	Ternlang: eine Systemprogrammiersprache mit Compiler und Laufzeit auf Basis balancierter ternärer Logik, dazu ein Rahmenwerk für sparsame KI, das Modell albert. und ein Server für KI-Werkzeuge. Fallstudie in Kapitel 3.9.1.	LGPL-3.0, Rust	01.04.–13.11.2026, 30 Sterne
albert-spores	Sammelstelle, über die Beitragende das Modell albert. auf eigenen Rechnern mittrainieren und Zwischenstände (Spores) einreichen.	keine Lizenz angegeben, HTML	15.05.–24.07.2026, 2 Sterne
agent-albert-cli	Kommandozeilen-Client für albert. und weitere Sprachmodelle, Teil des Stacks.	keine Lizenz angegeben, Rust	22.06.–14.08.2026, 0 Sterne
bifp	Bidirectional Instant Feedback Protocol: Entwurf eines Protokolls für wechselseitige Rückmeldungen.	keine Lizenz angegeben, Rust	02.07.–24.07.2026, 0 Sterne

Systeme: ternäre Logik in jeder Schicht

REPOSITORY	WAS ES IST	LIZENZ LAUT GITHUB, SPRACHE	ZEITRAUM, STERNE
rusty-penguin	Betriebssystem für x86_64 komplett von Grund auf in Rust, mit eigenem Kernel, Treibern, Fenstersystem und Anwendungen, ohne Linux-Kernel. Fallstudie in Kapitel 3.9.2.	MIT, Rust	26.05.–19.10.2026, 18 Sterne

Verlässliche KI: prüfen, messen, begrenzen

REPOSITORY	WAS ES IST	LIZENZ LAUT GITHUB, SPRACHE	ZEITRAUM, STERNE
gut-check	Prüftor für KI-Agenten mit zwei Schleifen: Ein schneller Klassifikator markiert Unsicherheit und reicht sie an das große Sprachmodell weiter.	Apache-2.0, Python	22.09.–06.10.2026, 0 Sterne
ai-security	Sammlung von Werkzeugen zur Sicherheitsprüfung von KI-Systemen: Red-Teaming, Prompt-Injection, Leitplankentests und Agentensicherheit.	MIT, Python	24.09.2026, 0 Sterne
call-laura	Deterministischer Prüfserver für KI-Assistenten, aufbauend auf Lauras Forschungsrahmen zur Mensch-KI-Koevolution.	Lizenztext liegt vor, GitHub erkennt ihn nicht, Rust	12.07.–14.08.2026, 1 Stern

veo-framework	Technik, die ein Sprachmodell in einen reflektierenden Modus führt, aufbauend auf Lauras Forschungsrahmen.	MIT	19.07.–14.08.2026, 0 Sterne
laura	Köder-Datensatz (Canary-Token) gegen Datendiebstahl über NFC- und Bluetooth-Nähe, ohne Exploit und ohne Gerätezugriff, mit menschlicher Prüfung.	MIT, Python	17.07.–19.08.2026, 1 Stern
lauras-port-proxy	Offline installierbare Prüfung offener Netzwerk-Ports als PWA, ohne vorgetäuschtes Scannen.	MIT, HTML	17.07.–24.07.2026, 1 Stern
lauras-agents-public	Öffentliche Übersicht und Konzept der Agenten. Der Quellcode der Agenten bleibt im privaten Repository.	Lizenztext liegt vor, GitHub erkennt ihn nicht, Shell	16.07.–14.08.2026, 1 Stern

Offenlegung und Produkte

REPOSITORY	WAS ES IST	LIZENZ LAUT GITHUB, SPRACHE	ZEITRAUM, STERNE
rfi-irfos-web	Die Institutsseite rfi-irfos.com mit dem öffentlichen Verzeichnis der Untersuchungen und den PDF-Berichten. Fallstudie in Kapitel 3.9.3.	keine Lizenz angegeben, TypeScript	22.06.–16.11.2026, 1 Stern
.github	Organisationsprofil auf GitHub.	–	28.05.–24.07.2026, 0 Sterne

Anwendungen, Wissen und Experimente

REPOSITORY	WAS ES IST	LIZENZ LAUT GITHUB, SPRACHE	ZEITRAUM, STERNE
pass-the-bot	Zeigt, wie ein Bewerbermanagement-System einen Lebenslauf liest: Stellenanzeige einfügen, Lebenslauf hochladen, transparenten Stichwort-Abgleich erhalten, nach Angabe des Projekts ohne Konto und ohne Speicherung.	MIT, Python	14.09.–10.11.2026, 4 Sterne
invisible-layer	Webseite mit 44 Sensor-Experimenten, die zeigen, was ein Telefon alles messen kann.	keine Lizenz angegeben, JavaScript	23.05.–06.10.2026, 2 Sterne
ginie	Offline-Desktop-Begleiter und KI-Assistent, der von einem USB-Stick läuft (GTK3, portables Ollama).	keine Lizenz angegeben, Python	05.06.–06.10.2026, 2 Sterne

claude-cache-idiocracy	Werkzeug, das beim Arbeiten mit KI-Programmierassistenten den Verbrauch senkt: Es vermeidet doppelte Lesezugriffe, schützt Zwischenspeicher und verdichtet lange Sitzungen.	MIT, Python	24.–26.10.2026, 1 Stern
foodchain-analysis	Agentenbasiertes Simulationsmodell zur strukturellen Nahrungsknappheit mit Dashboard im Browser. Die Ergebnisse sind Aussagen des Modells.	keine Lizenz angegeben, HTML	17.06.–15.08.2026, 1 Stern
nikoletta-tutor	Website für ein Sprachcoaching, gebaut mit dem eigenen Baukasten samt Bearbeitungsbereich für die Inhaberin.	keine Lizenz angegeben, TypeScript	15.06.–24.07.2026, 0 Sterne
e-techbike-at, aladdin-mini, p2p-repository	Diese drei öffentlichen Repositories sind in diesem Bericht bewusst nicht beschrieben.	–	–

3.10.2 Private Repositories nach Aufgabe

Private Repositories zählen wir und ordnen sie einer Aufgabe zu. Ihre Einzelheiten bleiben privat, weil sie Kundenumgebungen, laufende Untersuchungen oder entgeltpflichtigen Quellcode berühren.

AUFGABE	ANZAHL	WAS DORT LIEGT
Produkte	2	Das Kundenportal des Managed SOC (Kapitel 3.2) und die Plattform The Squad für Sicherheitsuntersuchungen mit Agenten und menschlicher Prüfung. Beide sind entgeltpflichtig, deshalb ist der Quellcode privat.
Lighthouse	2	Interne Beobachtung von Reichweite und Auffindbarkeit. Die Werte daraus stehen in diesem Bericht als interne Messung (Kapitel 3.2).
Forschungslabor und Cockpit	4	Das Forschungs-Cockpit mit der Whitebox-Ansicht (Kapitel 3.8.3) und Labore für Mensch-KI-Interaktion und die Zuverlässigkeit von Belegen.
Prüf- und Untersuchungswerkzeuge	3	Das Untersuchungs-Werkzeug mit Beweisklassen (Kapitel 3.8.1), ein Prüftor für Veröffentlichungen und Material früherer Android-Prüfungen.
Agenten und Automatisierung	3	Agenten-Kern, Skill-Rahmenwerk (Kapitel 3.8.4) und ein Werkzeug für die Vertriebsanalyse.
Kommunikation	2	FluxPost für die Planung der Social-Media-Beiträge (Kapitel 3.9.5) und ein Werkzeug zur Erzeugung von Bildmaterial.
Sonstige	2	Ein Archivstand von 2025 und ein Entwurf vom November 2026.

3.10.3 Forks und Archiv

18 öffentliche Repositories sind Forks fremder offener Projekte oder frühe Fassungen von Ternlang. Die Urheberschaft liegt bei den Quellprojekten. Sie stehen hier der Vollständigkeit halber und zählen in den Kennzahlen dieses Berichts nicht als eigenständige Leistung (die 68 Sterne in Kapitel 3.2 beziehen sich auf die 23 eigenständigen Repositories).

REPOSITORIES	INHALT	LIZENZ LAUT GITHUB
13, genesis, ternary-intelligence-stack-genesis-archive (archiviert)	Frühe Fassungen und Archivstand von Ternlang, als Fork geführt (Vorläufer des Stacks, angelegt ab 03.09.2025).	Apache-2.0 bzw. LGPL-2.1 bzw. –
auto-bayesian, causal-perception-implementation, mutatis-mutandis, gen-fraud-graph, sota-stressed-datasets	Forschungscode zu Bayesschen Netzen, kausaler Wahrnehmung, Diskriminierungsanalyse, synthetischen Betrugsgraphen und Robustheits-Datensätzen.	Apache-2.0 (Datensätze: von GitHub nicht erkannt)
autoguardrails, mech-gov-framework, llm_bridge, linear-adapter-trainer, genetic-algorithm	Bausteine für KI-Leitplanken, mechanische Governance von LLM-Entscheidungen, einen einheitlichen LLM-Zugriff, Embedding-Adapter und genetische Suche.	Apache-2.0
ralph, ralph-vault-skill	Eine Schleife, die einen KI-Programmierassistenten je Durchlauf mit frischer Sitzung startet, und ein Skill für das zugehörige Wissensarchiv.	Apache-2.0
RuVector, laya, ntpsec	Vektor-Speicher mit Graph-Netz in Rust, ein Klassifikations-Rahmenwerk und der Spiegel einer Implementierung des Zeitprotokolls NTP.	MIT, Apache-2.0, von GitHub nicht erkannt

3.11 Veröffentlichte Papers

Neben dem Quellcode hat das Institut sechs Papers veröffentlicht. Veröffentlichungsort ist das Open Science Framework (OSF), eine Ablage für offene Forschung; jedes Paper trägt eine DOI und lässt sich dauerhaft zitieren. Die Reihenfolge folgt dem Veröffentlichungsdatum. Die Papers sind unser eigener Beitrag zur Fachdiskussion. Ihre Abstracts stehen im englischen Originalwortlaut; Zahlen und Aussagen darin sind Angaben der Autorinnen und Autoren und gehören nicht zu den Messungen dieses Berichts. Stand der Angaben: 17.11.2026.

Sechs Papers auf dem Open Science Framework, Juli 2025 bis Juli 2026



Abbildung 16: Quelle: OSF-Schnittstelle, Abruf 17.11.2026, Datum des OSF-Eintrags. CALL LAURA und JARVIS wurden am selben Tag veröffentlicht und teilen sich einen Punkt.

PedalGate v1.0: Dispatch Disparity, Platform Gaslighting and the Collapse of Courier Trust in Austria's Gig Economy

25.07.2025 · doi.org/10.17605/OSF.IO/H5U8F

EXECUTIVE SUMMARY IM WORTLAUT

Between April 15 and July 25, 2025, a comprehensive 101-day multi-platform investigation was conducted by RFI-IRFOS into systemic inequities, deeply rooted algorithmic dysfunctions, and patterns of rider classification exploitation within the Austrian gig economy. This research concentrated upon three of the nation's preeminent delivery platforms—Foodora, Volt, and Lieferando—which collectively constitute the preponderant share of the country's app-mediated logistics labor market. The study was predicated not upon isolated anecdotal accounts but upon a structured, recursive field research methodology, incorporating direct testimonies, meticulous behavioral and dispatch observation, exhaustive escalation documentation, and formal correspondence with designated platform communication representatives. The inquiry revealed a landscape wherein the ostensible promise of flexible employment frequently degenerates into a precarious existence, characterized by a profound and discernible power imbalance, systematically favoring the platform entities over the individual labor providers. This imbalance is observed to manifest through the deliberate obfuscation of operational mechanisms and the strategic manipulation of incentive structures.

The findings delineate a disturbingly consistent pattern: platforms are observed to be optimizing for opacity whilst surreptitiously entrenching asymmetry. Every constituent element of the logistics process—ranging from order assignment to bonus incentive logic, and extending even to the definitional parameters of a "vehicle"—is demonstrably skewed towards non-transparency and misclassification. This deliberate absence of clarity permits platforms to operate with minimal accountability, thereby transferring inherent operational risks and responsibilities onto individual couriers. The resultant economic precarity for riders is not merely an unfortunate byproduct but appears to be an intrinsic outcome of system design. Through the deployment of sophisticated linguistic constructs, selectively disclosed policy stipulations, and automated systems demonstrably lacking in human oversight or effective redress mechanisms, a feedback loop has been meticulously engineered whereby riders are compelled to bear the consequences of systemic flaws, technical anomalies, and intentional design choices. Concurrently, they are systematically denied access to the operational levers that directly impinge upon their remuneration,

occupational safety, and long-term economic viability. Riders are concurrently subjected to management by inscrutable code and held culpable for non-conformance to regulations that are neither overtly visible nor equitably enforced, thereby fostering an environment characterized by profound mistrust, pervasive economic insecurity, and a palpable sense of disempowerment amongst the labor force.

Bezug zu unserer Arbeit: Eigenständige Feldforschung zur Plattformarbeit, getrennt von den Untersuchungsberichten in Kapitel 3.9.3.

Operational Blueprint for Physical AI Deployment: An Ethics-First Architecture for Robust and Reproducible Physical AI Agents in Open-World Environments

27.07.2025 · doi.org/10.17605/OSF.IO/M3VG6

ABSTRACT IM WORTLAUT

Artificial-intelligence agents that roam the messy, open-world physical realm must juggle sensor noise, moral ambiguity, and their own fallibility—yet most contemporary systems still cling to brittle binary logic and opaque heuristics. We present the Operational Blueprint for Physical AI Deployment, an architecture that welds multimodal perception, SQL-backed episodic-semantic memory, a ternary decision algebra (-1 / 0 / +1), and an introspective learning loop into a single, field-ready organism. Our design formalises confidence propagation, introduces two novel diagnostic metrics (μDP and MDPi), and bakes ethical weighting ahead of optimisation. To ground rhetoric in reality, we execute simulated and field pilots across four domains—courier dispatch, mobile science, drone monitoring, and factory advisory—plus controlled benchmarks in OpenAI Habitat under escalating sensor-dropout chaos. Results show a 22% reduction in safety-related incidents and 15% faster task completion versus a binary-logic baseline, while maintaining GDPR-grade privacy. By openly specifying interfaces, latency budgets, and threat models, we aim to push physical-AI discourse from hype to reproducibility.

Bezug zu unserer Arbeit: Ternäre Entscheidungslogik für Agenten. Der Stack in Kapitel 3.9.1 setzt die ternäre Logik als Rechenbasis um.

The Ternary Intelligence Stack: An End-to-End Balanced-Ternary Alternative to Binary AI Infrastructure

01.04.2026 · doi.org/10.17605/OSF.IO/CYN28

ABSTRACT IM WORTLAUT

We present Ternlang, the first complete software stack for balanced ternary computing: a domain-specific language, bytecode compiler, stack-based virtual machine (BET VM), hardware description language backend, distributed actor runtime, and machine learning inference kernels—all unified under a single coherent architecture. The foundational primitive is the trit $t \in \{-1, 0, +1\}$, where the value 0 represents an active neutral state rather than absence, enabling three-valued logic that is structurally superior to binary for ambiguity-aware reasoning. The principal contribution is TSPARSE MATMUL: a first-class VM opcode that elides multiply operations against zero-weighted ("hold") trit elements, surfacing at the instruction-set level a property that BitNet-style [1] ternary quantization reveals in neural weights. Empirical evaluation on quantized weight matrices of 512 × 512 elements yields 56.2% sparsity and a 2.27× reduction in multiply operations versus dense execution; a Compressed Sparse Column kernel with Rayon parallelism achieves up to 122× speedup at 99% sparsity— without approximate arithmetic or hardware reconfiguration.

We further define the Balanced Ternary Execution (BET) ISA, a formal 2-bit-encoded instruction set with 51 opcodes spanning arithmetic, tensor operations, actor messaging, and control flow; synthesise it to Verilog-2001 with per-cell clock-gating on zero-weight elements; and demonstrate interoperability with existing ternary computing efforts via the ternlang-compat bridge crate.

Beyond the execution substrate, we introduce two higher-level contributions. The Ternary AI Reasoning Toolkit (Phase 8) provides five structural primitives—iterative deliberation, coalition voting, multi-dimensional action gating, scalar-to-temperature bridging, and hallucination scoring—that make any AI agent's decision loop architecturally ternary rather than merely prompted. The MoE-13 Ternary Orchestrator (Phase 9) implements a Mixture-of-Experts

head with dual-key synergistic routing, 1+1=3 emergent triad synthesis, a hard-wired safety veto, and a three-tier memory mesh, realising the MoE-13 architecture [2] in 177+ passing tests across the full workspace. All nine crates are published to crates.io under open-core licensing. A companion corpus of 28,500+ executable .tern programs demonstrates the hold state across domains spanning aerospace, medicine, distributed systems, civic governance, and AI agent design.

Version 2.0.0 of the Ternary Intelligence Stack introduces three further empirical advances. First, Albert MoE-13—a 22M-parameter fully ternary Mixture-of-Experts language model—demonstrates autonomous architectural self-evolution via Net2Net safe-copy layer surgery (3L→5L without restart), achieving an all-time best single-batch cross-entropy of 2.1353 (perplexity ≈8.5) trained entirely on a 2013 consumer CPU with no GPU. Second, independently verified AVX2 SIMD hardware benchmarks confirm 1.80× throughput at 50% sparsity and 3.0× over INT8 at 90% sparsity on a production Axum server. Third, the BET VM has been compiled to WebAssembly and executes live in the browser via TernStudio, proving that the complete ternary execution pipeline—lexer, parser, codegen, VM—runs client-side on standard web hardware without installation. The live MCP surface has grown to 34 free tools. These results establish ternary computation as a practical, democratising inference substrate rather than a theoretical curiosity.

The ongoing Albert MoE-13 v3.0 training programme extends these results with four further contributions. First, a vocabulary surgery expanded the model from an 8,000-token English-only tokenizer to a 32,000-token ByteLevel BPE vocabulary covering eight languages (EN, DE, FR, ES, TR, RU, AR, ZH), trained on a 177M-token corpus with a mandated 10% chaos layer. Second, autonomous depth expansion continued through thirteen sequential Net2Net surgeries from 5L to 26L, followed by an autonomous mycelial cord surgery that bifurcated the single 256-dimensional residual stream into a dual-stream 2×256H architecture fused by six anastomosis gates at Fibonacci-indexed layers [2, 3, 5, 8, 13, 21]—to our knowledge the first dual-stream ternary Mixture-of-Experts grown autonomously mid-training—reaching ≈187M parameters across 2,044 tensors while preserving all prior weight structure. Third, the Epoch-Gated Routing Lottery protocol resets all gate weights to kaiming-uniform at every epoch boundary while preserving expert weights, driving per-epoch routing competition and observed expert specialisation dynamics (TLIGHT: first Green experts at Global Epoch 56). Fourth, live inference benchmarks independently verified on two CPU-only machines confirm 75% expert skip per decode step (@sparseskip at 3/12 active experts) sustaining 79.5–92.2 tok/s (AMD Ryzen 5 PRO 3500U: 79.5 tok/s; Intel i7-4800MQ: 92.2 tok/s), with held-out WikiText-2 perplexity of 2026.1 on the v2.0.0 checkpoint, proving the ternary MoE substrate operates at conversational generation speeds without GPU hardware.

Bezug zu unserer Arbeit: Grundlagenpapier zum Repository ternary-intelligence-stack; Messwerte in Kapitel 3.2.

CALL LAURA: An Automatic Data-Processing Multi-Agent Framework (the "Agent SWAT Team" Methodology)

12.07.2026 · doi.org/10.17605/OSF.IO/QCVJB

ABSTRACT IM WORTLAUT

Paper 1 introduced two named instruments — the 8-Layer Cognitive Model and the User Integrity Protocol (UIP) — as ways of reading structure and integrity into a document or dialogue, and Paper 2 showed how a third instrument, the Continuous Co-Evolution Tracker, could be operationalized as mechanically computed software rather than left as self-reported narration. This paper documents a fourth operationalization, applied to a different problem: reviewing an arbitrary plan or document, once, and returning findings a reader can actually check. Existing document-review tooling built on language models suffers from one of two failure modes: either a single model call returns a blended verdict with no traceable link from a claim back to the evidence that produced it, or a multi-agent ensemble of model calls compounds the same problem across several unreadable reasoning chains, with no way for a user to tell whether any given finding reflects a named researcher's methodology or the tool's own invention. We describe call-laura, a small, deterministic multi-agent framework — an "agent team," each agent a pure function of the submitted text — that applies Paper 1's 8-Layer Model and UIP faithfully, adds two further lenses of the project's own, and attaches to every single finding a mandatory, type-enforced provenance tag stating exactly whose methodology produced it: the primary author's own published framework, this project's own operationalization of a concept she names but does not herself specify an algorithm for, or an addition with no connection to her work at all. We further

describe its extension, a fifteen-agent domain-specialist orchestration layer ("the SWAT team") that runs the same evidence-anchored, attribution-tagged discipline across a much wider set of concerns — security, legal and regulatory exposure, finance, operations, accessibility, and more — synthesizing convergent findings across agents into a single prioritized review rather than fifteen disconnected opinions. The central claim of this paper is narrower than it might sound: not that keyword and pattern matching understands a document better than a language model does — it does not, and we say so throughout — but that a review tool bearing a named researcher's identity has no integrity unless a reader can verify, per finding, whether a given claim is really her work or this project's own judgment call. Determinism is not offered here as a claim of superior insight; it is offered as the price of that verifiability being possible at all.

Bezug zu unserer Arbeit: Paper 3 der Reihe; gehört zu den Repositories call-laura und lauras-agents-public (Kapitel 3.10.1).

JARVIS: A Recursive, Self-Learning Human–AI Co-Evolution Agent Operating in Bidirectional Copilot Mode

12.07.2026 · doi.org/10.17605/OSF.IO/HC9ZB

ABSTRACT IM WORTLAUT

Paper 1 proposed the Integrated Emergent Interaction Architecture (IEIA-2025) and its Continuous Co-Evolution Tracker (CCET), and reported a Co-Evolution Index (CEI) rising from 0.31 to 0.82 across a single-subject, ≈3,800-session corpus — but, by its own §12 accounting, those figures were self-reported: narrated by the model within the dialogue it was describing, not computed by an independent pipeline. Paper 1 named this gap directly and called, in its Future Work (§13), for the metrics to be "standardized through shared benchmarking datasets and open-source analysis pipelines" before they could be treated as validated measurements rather than structured self-report. This paper documents the software system built to close that specific gap: Jarvis, a research agent embedded in the Emergent Interaction Lab's admin workspace that computes CEI, a companion Co-Evolution Points (CEP) measure, and a Resonance Frequency indicator mechanically, per turn, from stored message embeddings — the same embeddings used for the agent's cross-conversation retrieval, never recomputed, audited by automated integration tests against a real database rather than narrated in prose. We describe Jarvis's four coupled subsystems — the recursive co-evolution loop and its bidirectional copilot mode (the agent is bound to whatever module of the workspace the user is currently viewing, and may act on it directly, subject to a hard iteration cap and a rule that only a human ever publishes); a cross-conversation retrieval memory; the Continuous Co-Evolution Tracker itself; and an emergence-verification gate that promotes a qualitative observation to measured emergence only when it recurs across at least three independent sessions above a stricter similarity threshold than the one used for turn-to-turn stability. A cross-cutting attribution layer labels every output as structural adaptation, not autonomous cognition, and leaves missing data absent rather than invented. As in Paper 1, we are explicit about what this system does and does not establish. Reproducibility of computation is not validity of the underlying construct: a cosine-similarity threshold over embeddings is a mechanical, auditable proxy for "stability," not a certified measure of genuine semantic consistency, and its thresholds (0.75 for turn stability, 0.90 for cross-session emergence) are engineering choices tuned against one deployment, not empirically derived cutoffs. What this paper adds to Paper 1 is not a stronger scientific claim — it is the same claim, made checkable.

Bezug zu unserer Arbeit: Paper 2 der Reihe zur Mensch-KI-Ko-Evolution.

Human–AI Co-Evolution as Ternary Logic: Distinguishing Output, Emergent Signal, and Hallucination in Real Human–AI Systems

19.07.2026 · doi.org/10.17605/OSF.IO/UXCJE

ABSTRACT IM WORTLAUT

A recurring, unsolved problem in human–AI collaboration is that a system's routine output, a genuinely novel pattern emerging from real interaction, and a confident fabrication with no basis in fact can be indistinguishable in the moment, especially to a human working alongside the system in real time. This paper has two aims. First, it gives a self-contained account, for readers unfamiliar with either field, of what "emergence" means in the specific, weak,

> Bedau-sense this research program uses (§2), and of exactly how Papers 1–3 of this series measure it: the Continuous Co-Evolution Tracker's stability and Co-Evolution Index computation over stored turn embeddings, the two-stage emergence-verification gate that withholds "measured" status from a candidate signal until it independently recurs across sessions, and the parallel, human-side 8-Layer classification that keeps the measurement genuinely bidirectional rather than model-centric (§4). Second, it introduces Ternary Ground: a specified, not-yet-implemented proposal to represent the shared pending/verified/rejected structure already present, informally and in three mutually incompatible representations, across Papers 2 and 3's existing mechanisms, using RFI-IRFOS's independently developed Ternary Intelligence Stack (TIS) — a three-valued (trit, {−1, 0, +1}) computational substrate in which the zero state is a first-class, honest "hold," not a null, and in which every value can carry a structurally bounded, machine-readable confidence field (TritFloat) rather than an ad hoc float implying false precision (§5–6). We further document, narrowly and with the same transparency discipline as the rest of this series, that part of the impetus for this synthesis arose reflexively — during the very sessions spent validating and instrumenting the Co-Evolution framework, using tools whose own design this paper is partly about (§7). We are explicit throughout about what this paper does and does not establish: it gives a complete, checkable account of an existing measurement methodology, and it specifies a representational unification of three already-real mechanisms; it does not implement that unification in running code, does not improve the accuracy of any of the three mechanisms it describes, and does not resolve the harder question — left open by Papers 1–3 and inherited here — of whether any of these mechanical proxies actually track the theoretical constructs they are named for.

Bezug zu unserer Arbeit: Paper 4 der Reihe; verbindet die ternäre Logik mit der Prüfung von KI-Aussagen.

KAPITEL 4

Planung und Ausblick

Für jedes Ziel nennen wir die Messgröße, an der der nächste Bericht uns prüfen kann. Chancen und Risiken stehen mit ihren Gegenmaßnahmen nebeneinander.

4.1 Planung und Ziele

ZIEL	MESSGRÖSSE
Eigene Server so weit ausbauen, dass wir unser Sprachmodell lokal trainieren und die Forschung zur ternären Logik skalieren	Rechenkapazität der eigenen Server, Anzahl der Trainingsläufe auf eigener Hardware, neue Forschungsergebnisse zur ternären Logik pro Jahr
Zahlende Kunden finanzieren den Ausbau (Ziel: fünf bis 31.12.2026)	Anzahl monatlich zahlender Kundenverträge
Wirkung bei Kunden sichtbar machen (ab 2027)	Anzahl betreuter kleiner Betriebe, Anteil der Befunde mit bestätigter Korrektur (erkannt über den Merge im Portal), Zeit von der Meldung bis zur Korrektur
Klare Lizenzen für die eigenen Repositories	Anteil der eigenständigen öffentlichen Repositories mit Lizenz (heute 27 öffentliche Repositories mit Lizenz)
Wirkungsbericht als Jahresbericht	Veröffentlichung jährlich nach dem Social Reporting Standard
Forschung weiter öffnen	Neue öffentliche Berichte und Benchmark-Abschnitte pro Jahr

Geplant (Vision, noch ohne Umsetzung): Ein Aufforstungs- und Forschungsprojekt, das zerstörte Flächen wiederbelebt und Wissen darüber offen zugänglich macht. Dafür gibt es heute keine Mittel, wir nennen es nur als Richtung des Vereins.

4.2 Chancen und Risiken

Chancen

- Neue Vorgaben wie NIS2, der EU Cyber Resilience Act und der EU AI Act erhöhen den Bedarf an nachvollziehbaren Sicherheits- und Compliance-Nachweisen. Unser Kontrollkatalog deckt sie ab.
- Steigende Kosten für Rechenleistung machen ressourcenschonende Verfahren attraktiver.
- Offene Ergebnisse ziehen Forschungspartner und Mitwirkende an.

Risiken und Gegenmaßnahmen

RISIKO	GEGENMASSNAHME
Abhängigkeit von der Arbeit weniger Personen	Produkte finanzieren Betrieb und Forschung; Dokumentation und offene Repositories sichern Wissen im Verein
Zu wenig Nachfrage nach den Produkten	Preisstufe ab 99 € pro Monat, gezielte Ansprache kleiner Betriebe, offene Forschungsberichte als Vertrauensanker
Rechtliche und ethische Risiken der Sicherheitsforschung	Koordinierte Offenlegung, Berichte ohne personenbezogene Daten, Zugriff auf Kundenumgebungen nur mit ausdrücklicher Freigabe und eng begrenztem Umfang
Zu schnelle Wachstumsversprechen	Aussagen nur mit Messwert; Projektionen und Pläne als solche gekennzeichnet

KAPITEL 5

Organisationsstruktur und Team

Ein Verein, drei Co-Founder, ein Kassier und eine Gründungsgeschichte, die 2020 mit Studienfreunden begann.

5.1 Organisationsstruktur

Rechtsform: Verein, eingetragen im Zentralen Vereinsregister (ZVR-Zahl 1015608684), Entstehung am 13.11.2025, Sitz in Graz.

Vorstand und Vertretung: Obmann Simeon Andreas Kepp und Kassier Louis Paul Ehrig (Funktionsperiode 25.01.2026 bis 24.01.2030). Nach den Statuten (13.1) vertreten sie den Verein gemeinsam.

Zweck und Gewinnverwendung: Erforschung von KI-gestützten Zukunftstechnologien (Statuten 2.1). Die Tätigkeit ist nicht auf Gewinn gerichtet (2.2). Überschüsse dienen künftigen Vereinszwecken (3.2). Die Regeln zur Reinvestition (Monatsende, höchstens 10 % Rücklage, Jahresende bei null) beschreibt Kapitel 2.3.

Gewerbe: Der Verein ist selbst Inhaber des Gewerbes (Gewerbeanmeldung 25.01.2026, Dienstleistungen in der automatischen Datenverarbeitung und Informationstechnik). Die kommerziellen Produkte gehören dem Verein. Beteiligungen an anderen Unternehmen bestehen nicht.

Gründungsgeschichte: Am 17.06.2020 entschlossen sich Studienfreunde, gemeinsam Forschung zu betreiben. Aus einem Kollektiv wurde ein registrierter Verein in Vorarlberg. Nach dem Umzug nach Graz entstand der heutige Verein. Der Vorarlberger Verein wurde erst nach Abschluss aller Arbeiten aufgelöst. Das Institut wurde ohne Unterbrechung weitergeführt.

5.2 Vorstellung der handelnden Personen

PERSON	FUNKTION
Simeon Andreas Kepp MBA	Co-Founder, Obmann, gewerberechtlicher Geschäftsführer
Louis Paul Ehrig	Kassier
Zabihullah Karimi	Co-Founder, Kernteam
Laura Serna Gaviria	Co-Founder, Kernteam, Forschungspartnerin (Kapitel 6)

Simeon Kepp, Zabihullah Karimi und Laura Serna Gaviria haben das Institut gemeinsam gegründet und tragen die inhaltliche Arbeit. Zudem ist der Verein ein Kleinstunternehmen.

5.3 Partnerschaften, Kooperationen und Netzwerke

- **Forschungspartner:** Im Auftrag von Forschungspartnern untersuchen wir veröffentlichte mobile Apps (Kapitel 3.2).
- **Open-Source-Gemeinschaft:** Wir nutzen offene Werkzeuge (Semgrep, Gitleaks, Trivy) und veröffentlichen eigene Software unter freien Lizenzen.
- **Wirtschaftskammer:** Der Verein ist Mitglied der Wirtschaftskammer.

KAPITEL 6

Forschungspartnerschaft mit Laura Serna Gaviria

Laura Serna Gaviria ist Co-Founder und unsere Forschungspartnerin. Hier steht, was im Vereinsjahr auf ihrer Arbeit beruht.

6.1 Rolle und Zusammenarbeit

Laura Serna Gaviria hat das Institut mitgegründet und leitet das Emergent Interaction Lab. Seit 2023 erforscht Laura die Interaktion zwischen Mensch und KI über lange Zeiträume und hat dafür eigene Methoden entwickelt, mit denen sich kognitive, verhaltensbezogene und systemische Muster erkennen, rekonstruieren und bewerten lassen. Für das Institut arbeitet Laura als Forschungspartnerin an der Architektur, der Logik und der Prüfung unserer Agenten und der Plattformen, die auf ihnen laufen.

Dabei hat Laura sehr viele Stunden in das Institut investiert, und viele Ergebnisse dieses Vereinsjahres gibt es nur durch diesen Einsatz.

6.2 Was im Vereinsjahr auf Lauras Arbeit beruht

BEREICH	LAURAS BEITRAG	WO IM BERICHT, QUELLE
Agenten	Alle 292 Agenten mit der gesamten Logik der Multi-Agenten-Umgebung gebaut	Übersicht, Kapitel 3.9.4
The Squad	The Squad produktionsreif gemacht	Kapitel 3.2, Produkte
DINGIR, NYX und Sales-Intelligence-Agenten	Unermesslicher Beitrag zum Aufbau	NYX in Kapitel 3.8
Planung und Prüfung	Peer-Review jedes Workplans im Forschungsprogramm zum World Model seit etwa August 2026, in mehreren Runden	Aussage der Gründer, Planungsdokumente
Veröffentlichungen	CALL LAURA, JARVIS und Human–AI Co-Evolution as Ternary Logic beschreiben Software des Emergent Interaction Lab	Kapitel 3.11
Repositories	call-laura, lauras-port-proxy, laura, veo-framework und lauras-agents-public, zusammen 53 Commits	Kapitel 3.9.4

Quelle: Angaben der Gründer. Für die Peer-Reviews liegen Planungsdokumente vor, in denen Lauras Korrekturen übernommen sind. Ein Teil der Arbeit steht in Plänen, Konzepten und Reviews und taucht deshalb in den Commit-Zahlen aus Kapitel 3 nur teilweise auf. Eine Stundenzahl erfassen wir bisher nicht, der Bericht gibt darum keine an.

Anhang A: Zählweise und Grenzen

- **Datengrundlage.** Vollständige Historie der 41 eigenen Repositories (ohne Fork-Repositories) der GitHub-Organisation rfi-irfos am 16.11.2026, von denen 40 Commits im Berichtszeitraum enthalten und eines leer ist. Gezählt ist der Hauptzweig; zusätzliche Arbeit auf weiteren Zweigen bleibt unberücksichtigt (im Forschungs-Repository etwa 3.300 weitere Commits auf anderen Zweigen).
- **Commits.** 11.269 Commits im Hauptzweig seit 13.11.2025. Davon sind 152 reine Zusammenführungen von Zweigen. Textauswertungen (Betreffzeilen, KI-Kennzeichnung) beziehen sich auf die übrigen 11.117 inhaltlichen Commits.
- **Datum.** Das Datum ist das Commit-Datum der Historie. Innerhalb eines Tages ist die Reihenfolge nicht sicher ablesbar, zeitliche Aussagen gelten auf Tagesebene.
- **Betreffzeilen.** Ausgewertet wurden die Kurzbeschreibungen der Commits und die Versionsmarken, nicht der Inhalt der Änderungen. Eine Beschreibung belegt, dass ein Schritt so benannt wurde. Sie belegt nicht, dass die Funktion fehlerfrei läuft. Aussagen wie „works“ oder „complete“ sind Aussagen der Projekte.
- **Phasen.** Die Zeitabschnitte in den Fallstudien sind unsere Gliederung nach Datum und Inhalt. Die Projekte benennen keine Phasen.
- **Massenläufe.** Im Forschungs-Repository gelten als Massenlauf: Bausteine der Standardbibliothek (Präfix „stdlib“), automatische Kennzahl-Aktualisierungen, Daten-Patches und das Beobachtungsprotokoll des Trainings. Die Muster haben wir nach Durchsicht der Zeilen festgelegt und eher streng gewählt.
- **KI-Kennzeichnung.** Gezählt ist die Zeile „Co-Authored-By“ im Text des Commits; sie zeigt Beteiligung, nicht Umfang, und der Anteil von etwa einem Viertel KI-Arbeit (Kapitel 3.7) ist eine Schätzung der Gründer. Commits ohne die Zeile enthalten auch automatische Einträge (zum Beispiel aus dem Bearbeitungsbereich von Websites).
- **Autoren.** Die Historie enthält mehrere Kennungen für dieselbe Person. Eine Auswertung nach Person ist deshalb nicht möglich und hier nicht enthalten.
- **Reichweite.** Downloads und Sterne stammen von crates.io, HuggingFace und GitHub am 16.11.2026. Sie zeigen Interesse (Kundenwirkung: Kapitel 3.3 und 4.1).
- **Nicht enthalten.** Es sind keine Nutzer- oder Einspar-Zahlen aus den Commit-Nachrichten übernommen und keine Energiewerte in Watt. Private Repositories sind nur gezählt, nicht beschrieben.

Anhang B: Wörtliche Beispiele aus den Commit-Nachrichten

Die Beispiele erlauben, die Aussagen dieses Berichts in der Historie wiederzufinden. Der Wortlaut ist unverändert, die Sprache der Commits ist überwiegend Englisch.

DATUM	REPOSITORY	COMMIT-NACHRICHT
02.04.2026	ternary-intelligence-stack	Phase 5: Actor model — agent/spawn/send/await in ternlang
10.04.2026	ternary-intelligence-stack	chore(stdlib): purge 3000 hallucinated files, seed astro/ bench/ with real content
17.04.2026	ternary-intelligence-stack	release: v1.0.0 — first stable release
18.04.2026	ternary-intelligence-stack	feat(wasm): Phase 17 — real BET VM running in browser via WebAssembly
02.05.2026	ternary-intelligence-stack	docs(kpi): sanitize 122× throughput claims to transparent theoretical upper bounds
07.05.2026	ternary-intelligence-stack	chore(repo): VC/evaluator credibility sweep
11.05.2026	ternary-intelligence-stack	security: fix path traversal in stdlib_read and add allowlist to kpi_data
25.06.2026	ternary-intelligence-stack	feat(engram): ternary episodic memory — crate, MCP tools, per-key hosted store
24.10.2026	ternary-intelligence-stack	ci(explorer): build, test and deploy the explorer to GitHub Pages
23.10.2026	ternary-intelligence-stack	feat(explorer): structure/measured/illustrative labels
24.10.2026	ternary-intelligence-stack	feat(explorer): seven guided walkthroughs with prediction moments
24.10.2026	ternary-intelligence-stack	docs(explorer): live at albert.rfi-irfos.com
03.05.2026	ternary-intelligence-stack	feat(albert-training): upgrade to real-time websocket monitoring cockpit
13.05.2026	ternary-intelligence-stack	dashboard: stream watchdog + per-panel stale detection + GRAD norm fix
27.05.2026	ternary-intelligence-stack	obs: FN193 — Modal stall 18:26Z, 180s watchdog, clean stop, resume required
26.05.2026	rusty-penguin	feat(kernel): Phase 0 — bare-metal x86_64 kernel boots via GRUB, prints on VGA
29.05.2026	rusty-penguin	kernel: networking brick 4 — TCP + HTTP GET (the OS fetches a page)
30.05.2026	rusty-penguin	tls: from-scratch TLS 1.3 client — real HTTPS on bare metal (brick 7)

30.05.2026	rusty-penguin	kernel/sched: Increment 3c — ring-3 task in a private address space [UNVERIFIED]
02.06.2026	rusty-penguin	docs: add SESSION_LOG.md — honest per-session build record
22.06.2026	rfi-irfos-web	fix methodology language: decompilation and root-level comparison, not static APK analysis
26.06.2026	rfi-irfos-web	fix all R1 sent dates from Gmail + add 28 new ledger entries (113 clients)
30.06.2026	rfi-irfos-web	site: regulators in CC on every submission (was BCC)
09.08.2026	rfi-irfos-web	feat: prerender the SPA build so crawlers see real HTML content
12.08.2026	rfi-irfos-web	fix: correct app-count from unverified 315/649 to verified 311
21.08.2026	rfi-irfos-web	Compute corpus-depth KPIs from real report tree instead of hand-maintained numbers
31.08.2026	rfi-irfos-web	fix: reconcile ledger embargo dates with what was actually sent (Gate 3b)
31.10.2026	rfi-irfos-web	Atom feed of the 50 newest reports at /evidence/feed.xml
31.10.2026	rfi-irfos-web	A11y: skip-to-content link on the public site (EN/DE)
20.05.2026	albert-spores	feat: --contributor mode — auto-push spore on each checkpoint
22.09.2026	gut-check	feat: long multi-file task -- the hypothesis test, and it found a real failure
28.09.2026	pass-the-bot	feat: add honest 5-axis radar chart
21.10.2026	pass-the-bot	fix: show no score instead of 100% when a posting has no required skills
24.10.2026	claude-cache-idiocracy	Add fail-closed local proxy budget
17.07.2026	lauras-port-proxy	initial commit: honest offline port-checker PWA
19.07.2026	veo-framework	Add a real protocol for telling output/emergence/hallucination apart
05.10.2026	privates Repository (Untersuchungs-Werkzeug)	feat: add PQC zone, source ledger with fail-closed admission gate, crypto-use records and graph
06.10.2026	privates Repository (Untersuchungs-Werkzeug)	feat(investigate): M7 hash-chained ledger, probes and investigation loop with replay
06.10.2026	privates Repository (Untersuchungs-Werkzeug)	test(investigate): planted instruction in the artefact against an obedient model

02.11.2026	privates Repository (Untersuchungs-Werkzeug)	fix(soc-cases): unbypassable resolve gate, actor in events, savepoints
11.07.2026	privates Repository (Forschungslabor)	feat(safety): incorruptible-refusal instruction + Anomaly Watchdog v1 (#59) (#59)
18.07.2026	privates Repository (Agenten)	docs: add 'Human rights are not subject to negotiation.' as core doctrine subheader
04.09.2026	privates Repository (Skill-Rahmenwerk)	Add Authority matrix + Human Decision Gate (Gate 0)
01.11.2026	privates Repository (Forschungs-Cockpit)	feat(whitebox): clusters and objects explained in plain German
06.11.2026	privates Repository (Forschungs-Cockpit)	fix(eris): brain sends nothing out until the European route stands
07.11.2026	privates Repository (Forschungs-Cockpit)	fix(research): results count only with a logged second-model check; two empty runs stop the loop
16.05.2026	albert-spores	albert-train: open dashboard with CPU-safe poll and stale thresholds
07.08.2026	privates Repository (Agenten)	feat(core): Telemetrie pro Agent und frei waehlbarer LLM-Kopf
05.09.2026	privates Repository (Skill-Rahmenwerk)	Gate 7: Agent Health Dashboard + Drift Monitor (Spec 7.22)

Quellenverzeichnis

Abruf der externen Quellen am 16.11.2026.

- EY Österreich: Cybersecurity-Studie 2025, ey.com/de_at/newsroom/2025/09/ey-cybersecurity-2025 (32 % betroffene Unternehmen, 73 % Phishing, 200 Befragte ab 20 Beschäftigten).
- ISC2: 2025 Cybersecurity Workforce Study, isc2.org/Insights/2025/12/2025-ISC2-Cybersecurity-Workforce-Study (88 % mit Folgen durch fehlende Fähigkeiten, 16.029 Befragte, Erhebung Juli und August 2025).
- Weltwirtschaftsforum: Future of Jobs Report 2025, Pressemitteilung vom 08.01.2025, weforum.org/press/2025/01 (170 Mio. neue, 92 Mio. wegfallende Stellen, 41 % Arbeitgeber mit geplantem Personalabbau durch KI-Automatisierung, 59 von 100 mit Weiterbildungsbedarf).
- Internationale Energieagentur: Energy and AI, Executive Summary, iea.org/reports/energy-and-ai/executive-summary (415 TWh im Jahr 2024, rund 945 TWh bis 2030).
- KMU Forschung Austria, Schätzung 2022 (rund 601.300 KMU, 99,8 %), zitiert im Bericht an den Nationalrat, parlament.gv.at/dokument/XXVII/III/1123/imfname_1613206.pdf.
- Wirtschaftskammer Österreich: Größenklassenauswertungen, Definition der Größenklassen nach EU-Empfehlung, wko.at/statistik/kmu/gk-info.pdf.
- RFI-IRFOS: Forschungs-Repository ternary-intelligence-stack mit BENCHMARKS.md, github.com/rfi-irfos/ternary-intelligence-stack (öffentlich, LGPL-3.0).
- RFI-IRFOS: GitHub-Organisation rfi-irfos, Zählung am 16.11.2026 über die GitHub-Schnittstelle. Repositories ternary-intelligence-stack und rusty-penguin (README, Sterne, Lizenz). Pakete auf crates.io (Namensuche ternlang, albert, moe, lauras, bifp; Downloads, Abruf 16.11.2026). Modelle und Datensatz auf huggingface.co/rfi-irfos (Abruf 16.11.2026). Git-Historie aller eigenen Repositories (Hauptzweig, Auswertung 16.11.2026). Eigenmessung der Reichweite (internes Dashboard, Stand 16.11.2026).
- RFI-IRFOS: Commit-Historie privater Repositories (Forschungs-Cockpit, Untersuchungs-Werkzeug, Skill-Rahmenwerk, Agenten-Kernel, Forschungslabor) zu Whitebox und Sicherheitsrahmen; Auszüge auf Anfrage einsehbar.
- RFI-IRFOS: Xperience albert., albert.rfi-irfos.com (Live-Prüfung 16.11.2026). FluxPost-Export für Social Media, Zeitraum 30.10. bis 16.11.2026, Tageswerte der Plattform LinkedIn ab 04.08.2026.
- RFI-IRFOS: Vortrags-Codex „Das programmierbare Brot“ (Version 1.0, 2026), Sponsoring-Paket, Workshop-Unterlagen „AI zum Anfassen“ und „KI-Degustation“ (Institutsunterlagen, auf Anfrage einsehbar).
- RFI-IRFOS: Veröffentlichte Papers auf dem Open Science Framework (osf.io/rzvyg), DOI 10.17605/OSF.IO/H5U8F, M3VG6, CYN28, QCVJB, HC9ZB und UXCJE (Abruf 17.11.2026).
- RFI-IRFOS: Website rfi-irfos.com, Abschnitt „Our research, published“ (Stand 16.11.2026).
- RFI-IRFOS: Managed SOC portal.rfi-irfos.com (Dokumentation des Funktionsumfangs im privaten Repository, auf Anfrage einsehbar).
- Vereinsregister: Auszug ZVR 1015608684, Stichtag 16.11.2026. Statuten des Vereins (Fassung 2025).