

# Intelligente architecturen voor Rules as Code

**Rules as Code (RaC)** is de benadering om wetten en regels te representeren in **machine-leesbare code**, parallel aan de traditionele tekst. Dit maakt geautomatiseerde toepassing en validatie van regelgeving mogelijk. Recente ontwikkelingen gaan verder dan het louter codificeren van wetsregels: men onderzoekt architecturen die **intelligenter** en **semantisch rijker** zijn. Dergelijke architecturen combineren formele regels met AI-technieken, ontologieën en kennisgrafen om **contextbewustzijn** en **geavanceerde redenering** te ondersteunen. Hieronder bespreken we uiteenlopende architecturen en projecten die “*beyond Rules as Code*” gaan, inclusief hun inzet van semantische webtechnologie, AI, reasoning engines en adaptieve mechanismen.

## Integratie van semantische technologieën in RaC

([AI-Powered Rules as Code: Experiments with Public Benefits Policy - Digital Government Hub](#))

*Figuur: Conceptuele weergave van hoe wetgeving getransformeerd wordt naar gestandaardiseerde, machine-leesbare regels die kunnen worden ingezet in uitvoeringssystemen (Beeck Center, 2025).*

Een belangrijke trend is het verrijken van RaC-modellen met **semantische webtechnologieën**. Dit houdt in dat **juridische concepten en relaties expliciet worden gemodelleerd** met ontologieën en gelinkt aan wettelijke teksten. Zo’n aanpak maakt **geautomatiseerde semantische interpretatie** mogelijk. Een voorbeeld is de geïntegreerde architectuur geschetst door Monica Palmirani (Universiteit van Bologna), waar wetsteksten in een gestandaardiseerd XML-formaat (Akoma Ntoso) worden gekoppeld aan een **juridische ontologie** en formaliseringen in LegalRuleML (een XML-standaard voor regels) ([Diapositiva 1](#)). In deze architectuur bestaan drie lagen – **wetsdocumenten, juridische ontologieën en logische regels** – die onderling verbonden zijn. Dit creëert een rijke kennisstructuur waarin *reasoners* inconsistenties kunnen detecteren en conclusies kunnen afleiden op basis van zowel tekst als contextuele kennis ([Diapositiva 1](#)). Het koppelen van wetten aan **knowledge graphs** is ook in onderzoek. Het EU-project **Lynx** (2018–2020) bouwde bijvoorbeeld een *Legal Knowledge Graph* die wetgeving, jurisprudentie en standaarden integreert om slimme compliance-diensten te ondersteunen (). In Lynx worden juridische gegevens uit meerdere jurisdicties semantisch gelinkt; hierdoor kunnen bedrijven vragen stellen als “Wat is de equivalente regel X in land Y?” en via de kennisgraaf een antwoord krijgen (). Dergelijke semantische RaC-architecturen benutten **Linked Data**: wettelijke teksten krijgen stabiele ID’s (bijv. ELI voor wetten, ECLI voor uitspraken) en verwijzen naar ontologieën, zodat regels in code “weten” welke begrippen of definities in de echte wereld bedoeld zijn ([Diapositiva 1](#)). Dit maakt **geavanceerde bevraging en redenering** mogelijk; een regel-checker kan bijvoorbeeld via de ontologie externe data (bijv. IoT-sensorinformatie of andere datasets) betrekken bij het bepalen welke regels van toepassing zijn in een bepaalde context.

Een ander voorbeeld is het **Legal Knowledge Interchange Format (LKIF)**, ontwikkeld in het Europese ESTRELLA-project. LKIF definieerde een ontologie van basis juridische concepten (in OWL) gecombineerd met formele regels (in SWRL) ([Legal Knowledge Interchange Format - Wikipedia](#)). Dit diende als onderdeel van een generieke architectuur voor *legal knowledge systems*, waarin wettelijke kennis van verschillende bronnen uitgewisseld en geautomatiseerd verwerkt kon worden ([Legal Knowledge Interchange Format - Wikipedia](#)). Deze semantische rijkdom gaat verder dan louter “if-then” coderen: het maakt *juridisch redeneren* mogelijk, zoals het toetsen van consistentie tussen regelgeving en het ondersteunen van juridische argumentatie (bijv. via argumentatie-systemen als Carneades gekoppeld aan de regels). Kortom, integratie met semantische

technologieën zorgt ervoor dat **regels als code niet op zichzelf staan**, maar ingebed zijn in een breder juridisch kennissysteem – een noodzakelijke stap richting “*intelligent regulation*” waarin machines de *bedoeling* en *context* van regels beter begrijpen.

## AI-geïntegreerde Rules as Code (hybride AI)

Naast semantiek wordt ook **kunstmatige intelligentie (AI)** ingezet om RaC krachtiger te maken. Een prominente benadering is **hybride AI**, waarbij machine learning wordt gecombineerd met symbolische regels. Een EU-studie over wetgevingsinnovatie pleit voor een *hybride AI framework* om de kwaliteit en transparantie van wetgeving te verbeteren ( ). Hierin ondersteunen machine-learning-componenten de mensen, terwijl de uiteindelijke regels als code op een verklaarbare, symbolische manier worden beheerd. Zo’n hybride aanpak is met name veelbelovend in het juridische domein, “waar context, waarden en concepten cruciaal zijn” ( ) – iets dat puur data-gedreven AI vaak mist. Concreet zou een *machine learning* model patronen kunnen ontdekken of teksten helpen analyseren, waarna de resultaten worden omgezet in formele regels die door een regel-**reasoning engine** uitgelegd en toegepast kunnen worden. Deze combinatie voldoet ook beter aan eisen van menselijke controle en verklaarbaarheid onder o.a. de EU AI Act ( ).

Er zijn al **proof-of-concepts** waarin AI de RaC-cyclus ondersteunt. Een actueel voorbeeld (2024) is een reeks experimenten van het Beeck Center (Georgetown University) waarbij **Large Language Models (LLMs)** werden ingezet om uit beleidsteksten uitvoerbare regels te genereren ([AI-Powered Rules as Code: Experiments with Public Benefits Policy - Digital Government Hub](#)). In deze *Policy2Code* prototypen liet men GPT-4-achtige modellen juridische teksten (zoals criteria voor voedselsteun of Medicaid) omzetten naar logica in code. De bevinding was dat LLM’s dit deels kunnen, mits met zorg ge-**prompt** (gedetailleerde aanwijzingen) en dat menselijk toezicht onmisbaar blijft voor nauwkeurigheid ([AI-Powered Rules as Code: Experiments with Public Benefits Policy - Digital Government Hub](#)) ([AI-Powered Rules as Code: Experiments with Public Benefits Policy - Digital Government Hub](#)). Het toont wel aan dat **generatieve AI** kan helpen in de *RaC-pijplijn*: van ruwe tekst naar eerste codeversie, die vervolgens verfijnd wordt. Tegelijk biedt dit kansen om de regels als code weer te geven in *natuurlijk taalvorm* (voor verklaarbaarheid naar beleidsmakers en burgers).

Een ander AI-component is **Explainable AI (XAI)** binnen RaC. In Australië hebben Andrew Mowbray en collega’s de **DataLex**-benadering ontwikkeld, die expliciet inzet op symbolische AI met uitlegbaarheid ([Law as Code: Introducing AustLII’s DataLex AI by Andrew Mowbray, Graham Greenleaf, Philip Chung :: SSRN](#)). DataLex is een toolkit om wetgeving om te zetten naar beslisbomen en logica (via de eigen yscript-taal), inclusief een “**legislation pre-processor**” (ylegis) die wetsartikelen automatisch transformeert naar regels ([Law as Code: Introducing AustLII’s DataLex AI by Andrew Mowbray, Graham Greenleaf, Philip Chung :: SSRN](#)). Uniek is dat DataLex vrijwel volledige *transparantie* biedt: de logica blijft dicht bij de natuurlijke taal van de wetgeving, zodat zowel juristen als AI-systemen het kunnen volgen ([Law as Code: Introducing AustLII’s DataLex AI by Andrew Mowbray, Graham Greenleaf, Philip Chung :: SSRN](#)). Recente uitbreidingen koppelen dit aan XAI-principes – de redeneringen van de **rule engine** kunnen toegelicht worden in termen van wetsbepalingen ([Explainable AI \(XAI\) in Rules as Code \(RaC\): The DataLex approach](#)). Zo ontstaat een **verklaarbare beslissingsondersteuning**: bijvoorbeeld bij het geautomatiseerd toekennen van een overheidsuitkering kan het systeem exact aangeven welke artikelen en voorwaarden hebben geleid tot de beslissing, wat belangrijk is voor rechtvaardigheid en beroep. Dit sluit aan bij het idee van “*legale expert-systemen*” die reeds decennia worden onderzocht, maar nu in een modern RaC-jasje gestoken met AI-hulpmiddelen.

De combinatie van **ML en regels** wordt ook elders verkend. Zo zijn er onderzoeken naar het extraheren van regels uit beleidsdocumenten m.b.v. NLP ([Collaborative artificial intelligence system for investigation of healthcare claims compliance | Scientific Reports](#)) en het gebruiken van kennisgrafen om ML-modellen te voeden en te controleren ([Algorithmic disclosure rules | Artificial Intelligence and Law](#)) ([Algorithmic disclosure rules | Artificial Intelligence and Law](#)). Het overkoepelende doel is een soort “*slimme regelgeving*” te creëren, waarbij AI helpt bij het **analyseren, simuleren en zelfs optimaliseren** van regelgeving, terwijl de formele regels als code de **betrouwbaarheid en consistentie** waarborgen.

## Adaptieve en contextbewuste regelsystemen

Een volgende stap richting **intelligent regulation** is het ontwerp van regelsystemen die **adaptief** en **context-aware** zijn. Dit betekent dat de toepassing van regels dynamisch kan inspelen op de situatie van een gebruiker of op veranderende omstandigheden. In praktische zin zien we dit terug in concepten als de *Personal Regulations Assistant (PRA)* die in Nederland wordt verkend (). Het idee van de PRA is een persoonlijke digitale assistent die burgers helpt begrijpen welke regels en voorzieningen op hen van toepassing zijn. Om zo’n assistent te realiseren, volstaat het niet dat de wetten als code zijn vastgelegd – het systeem moet ook **gebruikerscontext** (persoonlijke gegevens, omstandigheden) meewegen en elke stap **uitlegbaar** houden (). RaC-architecturen voor zo’n toepassing bevatten daarom componenten voor **persoonlijk gegevensbeheer**, en een reasoning engine die regels dynamisch op de individuele situatie toepast. Belangrijk is dat alle afleidingen traceerbaar zijn naar officiële bronnen (regels *as code* én de juridische tekst) (). Dit stelt hoge eisen aan de architectuur: integratie met bijvoorbeeld **Self-Sovereign Identity** of persoonlijke datakluisen voor gebruikersdata, en het combineren van verschillende regelbronnen (gemeentelijk, landelijk, etc.) in één advies, terwijl conflicten geautomatiseerd opgelost moeten worden () ([Code-driven ‘law’ • COHUBICOL](#)).

Adaptiviteit komt ook naar voren in **beleidsimplementatie**. Bijvoorbeeld in Australië is een RaC-proof-of-concept gebouwd rond COVID-19 regels, waarbij een centrale regels-engine (OpenFisca) via APIs gekoppeld is aan verschillende front-ends (web, voice assistant) ([GovCMS Rules-as-Code Proof-of-Concept Whitepaper v1.2](#)) ([GovCMS Rules-as-Code Proof-of-Concept Whitepaper v1.2](#)). Dit systeem demonstreerde dat burgers via een voice interface (zoals Alexa) vragen kunnen stellen (“Moet ik nog in quarantaine volgens de huidige regels?”) en dat de achterliggende RaC-dienst het antwoord *contextspecifiek* genereert op basis van actuele regelgeving en persoonlijke input. De architectuur was microservice-gebaseerd, zodat bij **regelwijzigingen** slechts de centrale regelservice hoefde te worden geüpdatet en alle gekoppelde applicaties direct de nieuwe logica gebruikten ([GovCMS Rules-as-Code Proof-of-Concept Whitepaper v1.2](#)) ([GovCMS Rules-as-Code Proof-of-Concept Whitepaper v1.2](#)). Dit is essentieel voor **adaptiviteit**: regelgeving verandert regelmatig, en intelligente RaC-architecturen voorzien in mechanismen om nieuwe of gewijzigde regels *automatisch door te voeren* in allerlei digitale diensten. Sommige systemen kunnen zelfs proactief signalen afgeven – zogeheten “*predictive*” of “*preventive*” functies. Denk aan een compliance-systeem dat real-time sensordata (bijv. milieuwaarden bij een fabriek) toetst aan de normen en direct bijstuurt of waarschuwt als een drempel bijna overschreden wordt. Dergelijke koppelingen tussen **IoT en regels** zijn genoemd in het kader van slimme steden en smart contracts, waar sensoren acties kunnen triggeren die door wettelijke regels zijn toegestaan of verboden ([Diapositiva 1](#)). Hoewel dit grotendeels conceptueel is, tonen pilots aan dat contextbewuste regelhandhaving technisch mogelijk is.

In onderzoek naar **normatieve multi-agent systemen (MAS)** is adaptiviteit eveneens een thema. Hierin worden autonome software-agenten uitgerust met *normen* (regels) die hun gedrag sturen, en de MAS-simulatie onderzoekt hoe regels kunnen worden afgedwongen of geschonden ([View of](#)

[Dilemmas in Legal Governance](#)) ([Code-driven 'law' • COHUBICOL](#)). RaC draagt bij door die normen expliciet en computable te maken. Bijvoorbeeld, een MAS in het verkeer zou verkeersregels als code bevatten en “intelligent” kunnen omgaan met uitzonderingen (een ambulance-agent mag regels overtreden). Adaptive RaC-architecturen leren soms van omgeving of feedback – zogeheten *regulative feedback loops* – waarbij bijvoorbeeld een AI-agent signaleert dat een bepaalde regel contraproductief is in context X en dit terugmeldt aan beleidsmakers. Zo’n niveau van intelligentie staat nog in de kinderschoenen, maar concepten als “*regelgevende zelflerende systemen*” worden voorzichtig verkend onder banners als **smart governance**.

## Reasoning engines en juridische kennisystemen

**Reasoning engines** vormen het hart van veel Rules as Code-architecturen. Dit zijn de “intelligente” componenten die logische conclusies trekken uit de gecodeerde regels. Traditionele voorbeelden zijn *rule-based expert systems* en inferentiemachines (zoals Prolog, of modernere varianten als Datalog/ASP). In de context van RaC zien we uiteenlopende engines: van **berekeningen** (bijv. OpenFisca voor sociaalrechtelijke berekeningen) tot **logische afleiders** (zoals Blawx, dat met *Answer Set Programming* logisch kan redeneren over wetstoepassing) ([Blawx: Rules as Code Demonstration - MIT Computational Law Report](#)). De keuze voor een reasoning engine bepaalt deels hoe “slim” het systeem is. Een *deductieve* engine kan vragen beantwoorden als “Valt persoon X in aanmerking voor regeling Y?”, terwijl een *inductieve* variant misschien hypotheses kan testen (“Welke combinatie van omstandigheden levert recht op steun Z op?”). Sommige architecturen combineren meerdere reasoners: een ontologie-reasoner (voor conceptuele relaties) naast een regels-reasoner (voor besluiten).

Daarnaast zijn er **intelligente agenten** die in regelgeving worden ingezet om automatisering te bereiken. Een bekend concept is “*regels als agenten*” – regels gedragen zich als autonome objecten die signaleren wanneer ze van toepassing zijn en interageren met andere regels. Hoewel dit meestal theoretisch is, hebben projecten als **IBM’s Watson voor wetgeving** en diverse EU-initiatieven geëxperimenteerd met agent-gebaseerde simulaties van regelgeving (bijv. agents die in een simulatie de effecten van nieuwe wetgeving “uitspelen”). Zulke systemen vallen onder *Legal Knowledge Systems*, een onderzoeksgebied dat al decennialang bouwt aan **juridische AI**. Moderne RaC-architecturen leunen op die expertise: ze hergebruiken formalisme uit de AI&Law-wereld (ontologieën, deontische logica, argumentatiemodellen) om te zorgen dat regels-as-code niet slechts harde if-then instructies zijn, maar ingebed in een **kennisrijke, redeneerbare omgeving** ([Legal Knowledge Interchange Format - Wikipedia](#)).

Een praktisch voorbeeld is **Oracle Policy Automation** (nu Oracle Intelligent Advisor), een commercieel systeem dat door overheden wordt gebruikt om wet- en regelgeving te implementeren. Het volgt een RaC-achtige aanpak: beleidsregels worden in een beslissingsmodel vastgelegd en de engine voert automatisch de logica uit in diensten. Uniek is dat het ook **contextuele interviewflows** genereert: de burger krijgt alleen die vragen die relevant zijn gegeven eerdere antwoorden, een vorm van *context-aware branching*. Hoewel dit platform gesloten is, illustreert het de rol van geavanceerde reasoning: de motor kan uitleg geven (“*U komt niet in aanmerking omdat uw inkomen boven drempel X ligt volgens artikel Y*”) en kan omgaan met datumlogica, uitzonderingen en voorkeurregels – allemaal eigenschappen die een **juridisch kennisgestuurd systeem** maken.

Kortom, reasoning engines en slimme agenten zorgen dat een RaC-architectuur niet statisch is, maar als een **juridisch brein** fungeert binnen software. Ze voeren automatisch besluitvorming uit, checken naleving (compliance) en ondersteunen bij beleidsanalyses (“*Wat als we deze parameter wijzigen?*”). In combinatie met de eerdergenoemde semantische componenten en AI-ondersteuning ontstaat zo een nieuwe generatie **legal knowledge systems**: systemen die wetten niet alleen als data

zien, maar als **levende kennisregels** die op intelligente wijze toegepast en geanalyseerd kunnen worden.

## Voorbeelden van architecturen en projecten

Onderstaand overzicht illustreert bestaande (of voorgestelde) RaC-architecturen, de gebruikte technologieën en hun intelligentiecomponenten:

Project/ architectuur	Technologieën / aanpak	Intelligentiecomponenten
<b>AustLII DataLex (Australië)</b>	Yscript regeltaal & interpreter; <i>Ylegis</i> pre-processor die wets-XML omzet naar code ( <a href="#">Law as Code: Introducing AustLII's DataLex AI by Andrew</a> )	Symbolische AI (rule-based); <b>Explainable AI</b> uitlegmogelijkheden voor elke stap ( <a href="#">Explainable AI (XAI) in Rules as Code (RaC): The DataLex approach</a> ); schaalbaar
<b>Lynx kennisgraaf (EU)</b>	Semantische <b>kennisgraaf</b> van wetgeving, jurisprudentie en standaarden (gebaseerd op ontologieën,	<b>Semantisch redeneren</b> over compliance; verbonden data voor <i>smart queries</i> (multilingual vraag-antwoord over regels).
<b>OpenFisca &amp; GovCMS POC</b>	OpenFisca Python regels-engine als microservice; integratie met Drupal CMS via JSON API ( <a href="#">GovCMS Rules-</a>	<b>Rule-based simulatie</b> van wetgeving (belasting/uitkeringen); <b>contextuele</b> bevraging via front-end (aanpasbare
<b>Better Rules (Nieuw- Zeeland)</b>	Co-design van regelgeving in <b>mens- en machineleesbaar</b> formaat; gebruik van domeinspecifieke taal (bijv. decision tables of Markdown) voor	<b>Mens-in-de-loop</b> ontwerp; focus op semantiek en consistentie bij wetgevingstraject; maakt geautomatiseerde diensten mogelijk (voorbereidend op AI-
<b>Blawx (Canada)</b>	Browsergebaseerde grafische editor voor regels; gebruikt <b>Answer Set Programming (ASP)</b> solver (Clingo) als reasoning-engine ( <a href="#">Blawx: Rules as</a>	<b>Automatisch redeneren</b> over juridische scenario's (o.a. "wat-als" analyse); kan complexe logica oplossen met uitleg; faciliteert snelle prototyping van
<b>LLM Policy2Code experiment</b>	Gebruik van GPT-3/GPT-4 <b>Large Language Models</b> om ongestructureerde beleidsdocumenten om te zetten in pseudo-code en testcases	<b>Natuurlijke taal AI</b> voor regelinterpretatie; versnelt codificatie door suggesties te genereren; menselijk toezicht zorgt voor correctheid (hybride AI aanpak).
<b>Palmirani's AI- gedreven</b>	Integrale architectuur met Akoma Ntoso (juridische XML), LegalRuleML, ontologieën (OWL) en gekoppelde open	<b>Geïntegreerde reasoning</b> : wetstekst + context + logica → conclusies via engine; ondersteunt IoT- en smart contract-
<b>LKIF / ESTRELLA (EU)</b>	Legal Knowledge Interchange Format: core-ontologie + SWRL regels; koppeling met argumentatiesysteem	<b>Logische interoperabiliteit</b> tussen juridische kennisbanken; geautomatiseerde <b>argumentatie</b> op basis van formele regels

(Bronnen: zie referenties. Tabel toont diverse initiatieven en hun slimme componenten, van semantische kennisgrafen tot AI-ondersteuning.)

## Belangrijke initiatieven, experts en leiders

Innovatie op het snijvlak van regelgeving en AI vindt wereldwijd plaats, met enkele pionierende landen en instituten:

- **Nieuw-Zeeland** – *Better Rules Initiative*: Pionier in het concept van wetgeving co-design in code; het Service Innovation Lab (govt.nz) experimenteerde als eerste met "law as code" in

praktijk. Experts: Nadia Webster, Brenda Wallace en collega's die het eerste *Better Rules* rapport opstelden.

- **Australië** – Sterke overheids- en academische betrokkenheid. AustLII (Australasian Legal Information Institute) met prof. Graham Greenleaf en Andrew Mowbray ontwikkelde DataLex ([Law as Code: Introducing AustLII's DataLex AI by Andrew Mowbray, Graham Greenleaf, Philip Chung :: SSRN](#)). Overheidszijde: DTO en NSW hebben RaC-prototypes gedaan; ook CSIRO's Data61 onderzoekt regtech. Pia Andrews (voorheen NSW, nu Canada) was een bekende voorvechter van RaC.
- **Canada** – Actief gemeenschap rond RaC. Jason Morris (Computable Law, Blawx) demonstreert hoe open source tools ingezet kunnen worden ([Blawx: Rules as Code Demonstration - MIT Computational Law Report](#)). Overheidsorganisaties (bijv. Service Canada en Canadian Digital Service) verkennen RaC voor dienstverlening. Canada host ook een *Rules as Code Community of Practice*.
- **Europese Unie** – Academische projecten als *MIREL* en *Lynx* (Madrid/Zaragoza) leggen basis met kennisgrafen en NLP voor juridische compliance. Monica Palmirani (Italië) en Giovanni Sartor zijn toonaangevende onderzoekers op gebied van juridische ontologieën en AI-wetgeving ([Diapositiva 1](#)). De Europese Commissie verkent RaC via het Interoperable Europe initiatief ([\[PDF\] Assessment of Rules as Code methods concerning the goal of ...](#)) en heeft via OECD OPSI papers als "*Cracking the Code*" ondersteund. Ook in Nederland (Digicampus, Tilburg University) en Frankrijk (Etalab, OpenFisca-team) wordt geïnnoveerd.
- **Verenigd Koninkrijk** – The National Archives (legislation.gov.uk) experimenteerde met wetgeving als data. John Sheridan wordt vaak genoemd als een vroege proponent van het idee om wet- en regelgeving direct als computercode beschikbaar te stellen.
- **Verenigde Staten** – RegLabs en denktanks (bijv. Georgetown's Beeck Center) richten zich op **AI in regelgeving** voor o.a. sociale voorzieningen ([AI-Powered Rules as Code: Experiments with Public Benefits Policy - Digital Government Hub](#)). Daarnaast speelt de technologische industrie (zoals Oracle met zijn beleidssystemen en IBM Watson) een rol in het leveren van platforms die de kloof tussen wet en uitvoering overbruggen.

Tot slot is de **OECD (OPSI)** een internationale drijvende kracht: hun rapport "*Cracking the Code*" (2020) heeft wereldwijd bewustwording gecreëerd voor RaC als onderdeel van *smart governance*. Hierdoor is er nu een netwerk van overheden en experts ontstaan die gezamenlijk werken aan standaarden, pilotprojecten en kennisdeling op dit gebied (). We zien dat de samensmelting van **juridische expertise, semantische webtechnologie en AI** leidt tot een nieuw paradigma van "*intelligente regelgeving*". Hoewel veel van deze initiatieven nog in ontwikkeling zijn, wijzen ze de weg naar een toekomst waarin **wetgeving zowel door mensen als door machines begrijpelijk en toepasbaar is**, met alle voordelen van dien voor transparantie, efficiëntie en naleving.

**Bronnen:** De gebruikte informatie en voorbeelden zijn afkomstig uit recente onderzoeksartikelen, whitepapers en praktijkcases, zoals geciteerd in de tekst ([Law as Code: Introducing AustLII's DataLex AI by Andrew Mowbray, Graham Greenleaf, Philip Chung :: SSRN](#)) () ([AI-Powered Rules as Code: Experiments with Public Benefits Policy - Digital Government Hub](#)), en weerspiegelen de stand van zaken anno 2024–2025 in het domein van Rules as Code en legal AI.