

ErgoAI Frequently Asked Questions

[What is ErgoAI?](#)

[What is knowledge representation and reasoning \(KRR\)?](#)

[What is rule-based KRR?](#)

[What is KRR good for?](#)

[What is Ergo \(a.k.a. ErgoAI Reasoner\)?](#)

[What is ErgoAI Studio?](#)

[ErgoAI and X](#)

[What is the difference between Ergo and Flora-2?](#)

[What is the difference between Ergo and XSB?](#)

[How is ErgoAI different from expert system shells of the 80's?](#)

[Is ErgoAI open-source?](#)

[Will ErgoAI put me in the red?](#)

[Can ErgoAI talk to the outside world?](#)

[Which other Languages does Ergo interface with?](#)

[Which data sources/formats can ErgoAI read or import?](#)

[Even Python? Java? JSON? XML? RDF?](#)

[What documentation about ErgoAI is available?](#)

[Are there examples of using ErgoAI in various contexts?](#)

[Can ErgoAI talk to Web services?](#)

What is ErgoAI?

ErgoAI is a rule-based, enterprise-grade [KRR \(Knowledge Representation and Reasoning system\)](#) that is highly expressive and feature-full, and yet is easy to learn and use. It consists of [ErgoAI Reasoner](#) (also referred to as just *Ergo*), [ErgoAI Studio](#) that serves as an IDE (integrated development environment) for the reasoner, and *connectors* to external data sources and formats as well as *interfaces* to various programming languages. See the [technical architecture of ErgoAI](#) for more details.

What is knowledge representation and reasoning (KRR)?

[Knowledge representation and reasoning](#) is a branch of Artificial Intelligence (AI) in which decision-making is done by way of reasoning from knowledge about the domain of discourse and general background knowledge. The knowledge is represented by way of logic statements, and reasoning is done by way of the inference methods of that logic. Different KRR approaches mainly differ in their underlying logics and the reasoning methods they support.

What is rule-based KRR?

Rule-based KRR uses only facts and rules of the form IF X THEN Y (and some more complex ones) to represent knowledge, trading some expressivity for scalability. Despite the syntactic restrictions, this form of KRR is highly expressive, and its scalability makes it practical.

What is KRR good for?

KRR is best suited for decision support in cases where domain knowledge can be elicited directly from *subject matter experts* (SMEs) with reasonable effort or from knowledge represented in text, such as regulations, contracts, textbooks, and other documents. This knowledge is then codified using logical statements. In these situations, KRR is far superior to Machine Learning (ML) because it is *explainable*, *incremental*, has *high-precision*, does not depend on a large number of examples, and is *not subject to spoofing or training bias*. Explainability is particularly important in law, health care, regulation, taxation, and the areas where mistakes may result in loss of life and disasters.

What is Ergo (a.k.a. ErgoAI Reasoner)?

Ergo is a powerful and highly expressive reasoning engine used by ErgoAI. It is described in these tutorials: [ErgoAI Tutorial](#) and [Capturing Real World Knowledge in Ergo](#). A complete reference to ErgoAI's KRR language is found in [ErgoAI Reasoner User's Manual](#).

What is ErgoAI Studio?

It's an IDE for the Ergo reasoner, which provides a syntax-driven editor, debugging facilities, explanations for answers returned by the reasoner, and more. See the [ErgoAI Studio User's Manual](#).

ErgoAI and X

What is the difference between Ergo and Flora-2?

ErgoAI Reasoner is an enterprise version of [Flora-2](#). Originally, ErgoAI was proprietary and Flora-2 was open-source, but in Spring of 2023 the two systems were merged keeping the ErgoAI name. The resulting system was open-sourced and it is now a replacement for Flora-2.

What is the difference between Ergo and XSB?

[XSB](#) is a powerful and complex reasoning engine used by Ergo reasoner. XSB is very versatile and has many capabilities, but requires considerable expertise to use. Think of Ergo as a Cadillac sedan with its buttons and simple controls vs. XSB, an engine attached to a chassis and controlled by pulling multiple levers. Unless you are a mechanic who likes tinkering with complex engines, ErgoAI is a better option for you.

How is ErgoAI different from expert system shells of the 80's?

Expert system shells were early, ad hoc systems that tried to exploit the idea of rule-based reasoning. They were not based on any rigorous theory and the technology used was rudimentary. Since then the necessary theories were developed and the supporting technology has advanced by leaps and bounds, making the current crop of KRR systems practical. Among these systems, ErgoAI stands out with its expressiveness and unique set of capabilities. The gap between ErgoAI and those expert system shells is akin to that between modern computers and those of the 80's.

Is ErgoAI open-source?

Yes, with Apache license v. 2.

Will ErgoAI put me in the red?

ErgoAI is free of charge under the terms of the open-source Apache license v. 2.

Can ErgoAI talk to the outside world?

Absolutely. Ergo can query external sources and import data in various formats. It can also invoke code written in other languages and be invoked from several popular languages. These facilities are described in the [Guide to ErgoAI Packages](#). Various examples of how to use these facilities can be found in the [Examples Bank](#).

Which other Languages does Ergo interface with?

Ergo reasoner has a bi-directional interface to Java, Python, C/C++, and Prolog. Some other languages have Java and/or Python interfaces, which can be used to create bridges to Ergo.

Which data sources/formats can ErgoAI read or import?

ErgoAI can query SQL databases, SPARQL endpoints, Web services, and it can import and query RDF/OWL, JSON, CSV, and XML.

Even Python? Java? JSON? XML? RDF?

You bet! See above.

What documentation about ErgoAI is available?

Tutorials: [ErgoAI Tutorial](#) (an introduction to ErgoAI) and [Capturing Real World Knowledge in Ergo](#) (using Ergo KRR to represent a real world domain).

Manuals: [ErgoAI Reasoner User's Manual](#) (a language reference), [Guide to ErgoAI Packages](#) (interfaces to Java, Python; import of Json, XML, CSV, etc.), and [ErgoAI Studio User's Manual](#) (how to use the IDE).

Are there examples of using ErgoAI in various contexts?

Oh, yes -- see the [Examples Bank](#).

Can ErgoAI talk to Web services?

Of course. The package that provides this capability is described in the [Guide to ErgoAI Packages](#), Chapter "*HTTP and Web Services*."