

## **[Dean Horak, Stanford University, Artificial intelligence, 2011 \(septembre 2013 puis 2015\)](#)**

Analyse de Dean Horak sur LinkedIn en September 2013 dans le thread " Artificial intelligence research revives its old ambitions" :

[https://www.linkedin.com/groups/Artificialintelligence-research-revives-its-old-127447.S.274713995?view=&srctype=discussedNews&gid=127447&item=274713995&type=member&trk=eml-anet\\_dig-b\\_pd-ttl-cn&fromEmail=&ut=3NRm516qMAbBY1](https://www.linkedin.com/groups/Artificialintelligence-research-revives-its-old-127447.S.274713995?view=&srctype=discussedNews&gid=127447&item=274713995&type=member&trk=eml-anet_dig-b_pd-ttl-cn&fromEmail=&ut=3NRm516qMAbBY1) :

"Jean-Philippe and I spent the morning (his afternoon) demoing his system. I promised him I would report back to the group my impressions, so here they are (Jean-Philippe, feel free to correct me if I got any of the details wrong).

Jean-Philippe has created a very nice, easy to use and fairly extensive expert system. It seems to contain all the niceties you might need in such a system such as database connectors, web connectors, file I/O, automatic form generation, automatic generation of conversational style interaction, etc. Essentially everything you might see in an IDE for a conventional programming language. However, instead of building programs directly using a programming language, programs are built using a graphical decision tree architecture.

The graphical interface generates what appears to be a comprehensive logic programming language, reminiscent of Prolog. Assuming the language is "Turing complete" (and I have no reason to doubt this), Jean-Philippe's claim that his system can produce any program that can be produced using other "programming languages" without requiring any programming knowledge is technically valid, though I would expect that the resulting application would likely be much less efficient and unwieldy than the same application developed in a more concise language such as C or C++.

Given it's apparent roots in Prolog, it fully supports deductive reasoning. This system also includes a nice built-in feature that presents a description of how the result was derived, which I envision to be a great aid in debugging and verification of the application.

My overall impression is that this system, within certain domains, is a viable option. For instance, coupled with a VRU (voice response unit), the system would allow for the easy development of interactive voice response systems for customer support, vmail systems, menu systems, etc. It seems to be a good fit for diagnostic applications such as those that might be used by a technicians in diagnosing equipment. In other words, any domain where an expert-system might be a good fit.

The main take-away point from the demo is that this system looks to be a real, production quality product, useful within certain application domains such as described above. On the other hand, I would not want to try and attempt to create a self-driving car using it, though, as I noted before, being "Turning complete", it should technically certainly be possible.

The other take-away point is that unlike the hyped-up claims of some on these forums (who's name I will not mention), this system delivers what the author claims (though my definition of what AI is fundamentally differs from Jean-Philippe's definition).

I wish Jean-Philippe luck in marketing his product which he has very obviously invested a great deal of effort into and I encourage anyone who is looking to develop an expert-system style application for which this would be a good fit, to contact him for a demo as well.

Finally, we were able to overcome our French/English language gap fairly well, though I will quickly admit that Jean-Philippe's English is far, far better than my French :)

Au revoir."