

Agent-based Computing Architecture and Design/Runtime Platform

<!-- -->

Table of contents

1 Datasheet.....	2
------------------	---

1. Datasheet

Solution data	
Name	Agent-based Computing Architecture and Design/Runtime Platform
Result type	Platform or platform component
Description/functionality	<ul style="list-style-type: none"> The agent based methodology allows the introduction of self-organisation, autonomy, flexibility, and robustness in service-oriented architectures. Interoperability is supported by explaining how general service-oriented architectures could be extended with these properties. The Jack Intelligent Agent™ Development Environment (JDE) was adopted as a designing and runtime tool for the design and execution of Belief Desire Intention (BDI) agents and teams of such agents. In the context of ATHENA JDE was extended with a library for the integration with service-oriented architectures (Web Services). Additionally model to text transformations were investigated to integrate JDE with the Eclipse modelling framework and with Rational Software Modeller.
Benefits to interoperability	The methodology allows to introduce self-organisaion, autonomy, flexibiliy, and robustness in service-oriented architectures. Interoperability is supported by explaining how general service-oriented architectures could be extended with these properties.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	The design and execution platform is used in the design of the agent-based components of the demonstrators developed in ATHENA.
Standards compliance	-
Availability	Installed service/solution
License	-
Status	Stable production version

Requirements/dependencies	<ul style="list-style-type: none">• JACKJACK
Web references	-
ATHENA metadata	
Contact person	Klaus Fischer, DFKI
Contributors	DFKI
Provided by project/activity	<ul style="list-style-type: none">• A5 – Planned and Customisable Service-Oriented Architectures• A6 – Model-driven and Adaptive Interoperability Architectures
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none">• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure
Used in pilot	-
Deliverable providing evaluation	-