

MASSim2Dev

From simulation to development in MAS
Repast-JADE automatic code generation for
interaction protocols

MAS: Multi Agent Systems

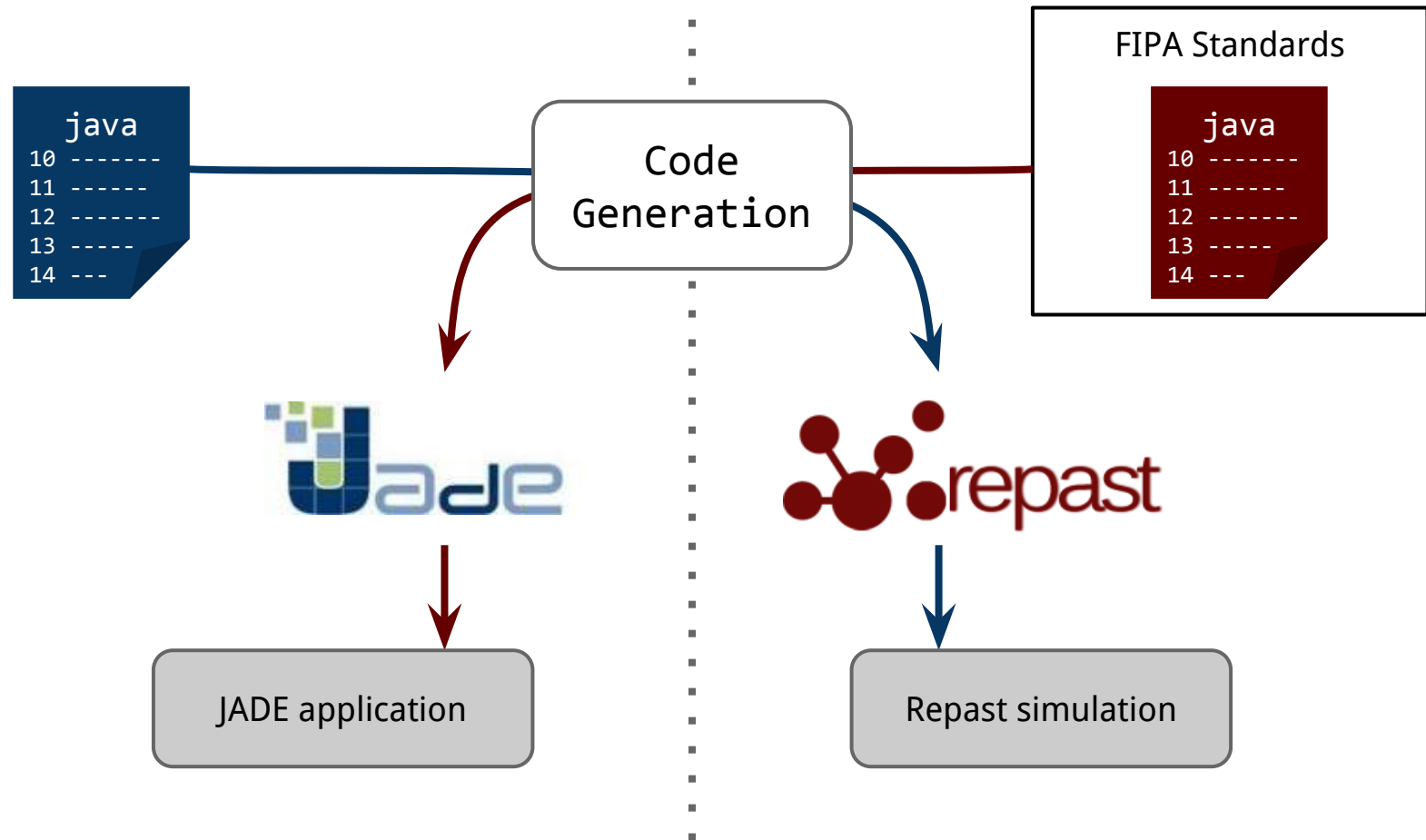
- » Intelligent agents: simple computing units cooperating or competing to achieve certain goals.
- » Frameworks help to code higher level applications and may provide the developer with many services.



JADE vs Repast

	JADE	Repast
Communication	FIPA-ACL Async	Shared resources Method invocation
Distributed System	Yes	No
Scalability	Limited	High
Agent Execution	Behaviour Based Multi Thread Event-driven (Async)	Schedule Based Single Thread Tick-driven (Sync)

Goal



<http://jade.tilab.com/index.html>

<http://repast.sourceforge.net/>

Goal

Code generation from JADE to Repast

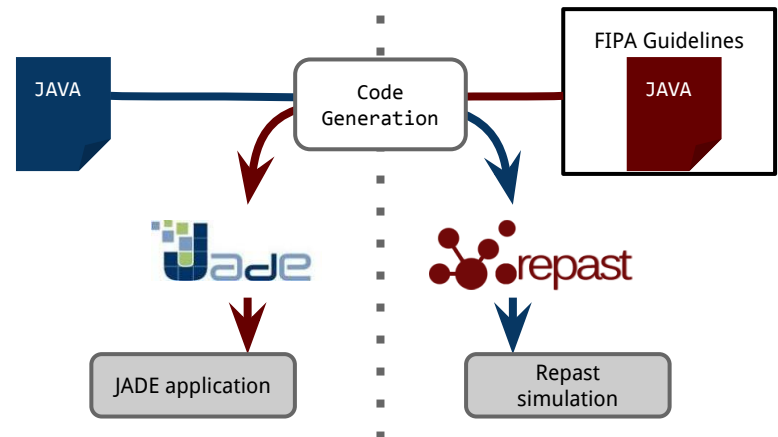
- » Simulation development
- » Statistical facilities
- » System scalability

Code generation from Repast to JADE

- » Multiagent application development
- » Communication protocols (FIPA)
- » Distributed system

Validation

- » Converted code retains functionality
- » Re-conversion must recover original code
- » Usability tests with developers



Related Work

JREP¹, MSIA²

- » Middleware between JADE and Repast Symphony
- » Agents represented as JADE, Repast, Middleware entities
- » Improved JADE performance for large number of agents
- » Uses Repast for simulation and visualization
- » Open platform allows integration with others

1) Jana Görmer, Gianina Homoceanu, Christopher Mumme, Michaela Huhn and Jörg P. Müller: JREP: Extending Repast Symphony for JADE Agent Behavior Components, IAT 2011.

2) Elena García, Sara Rodríguez, Beatriz Martín, Carolina Zato, Belén Pérez Lancho: MISIA: Middleware Infrastructure to Simulate Intelligent Agents, DCAI 2011, Salamanca, Spain, pp. 107-116.

Proposed solution

FIPA ACL³ support for Repast Symphony

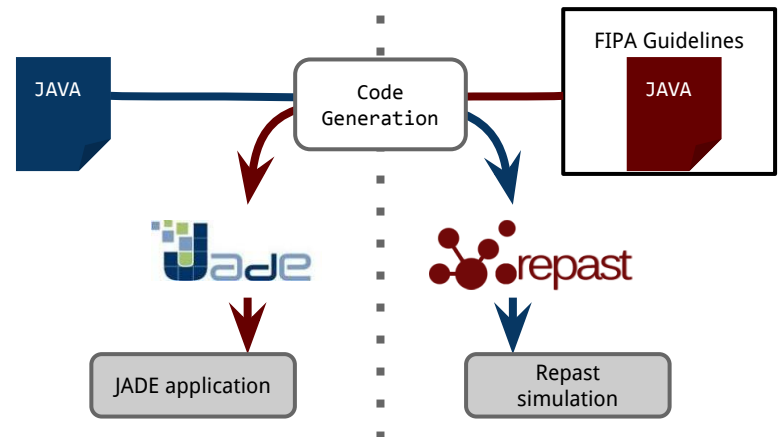
- » Implement communication library
- » Define usage guidelines

Use of Java analysis and refactoring

- » Java reflexion API
- » Spoon, Eclipse JDT

Preserve original code

- » Using this tool mustn't require extensive code changes (defeats the purpose!)
- » Data structures and lib calls must be mapped between frameworks.



3) <http://www.fipa.org/specs/fipa00061/index.html>

4) Pawlak, R., Noguera, C., Petitprez, N. Spoon: Program Analysis and Transformation in Java.

Code generation tools

Spoon

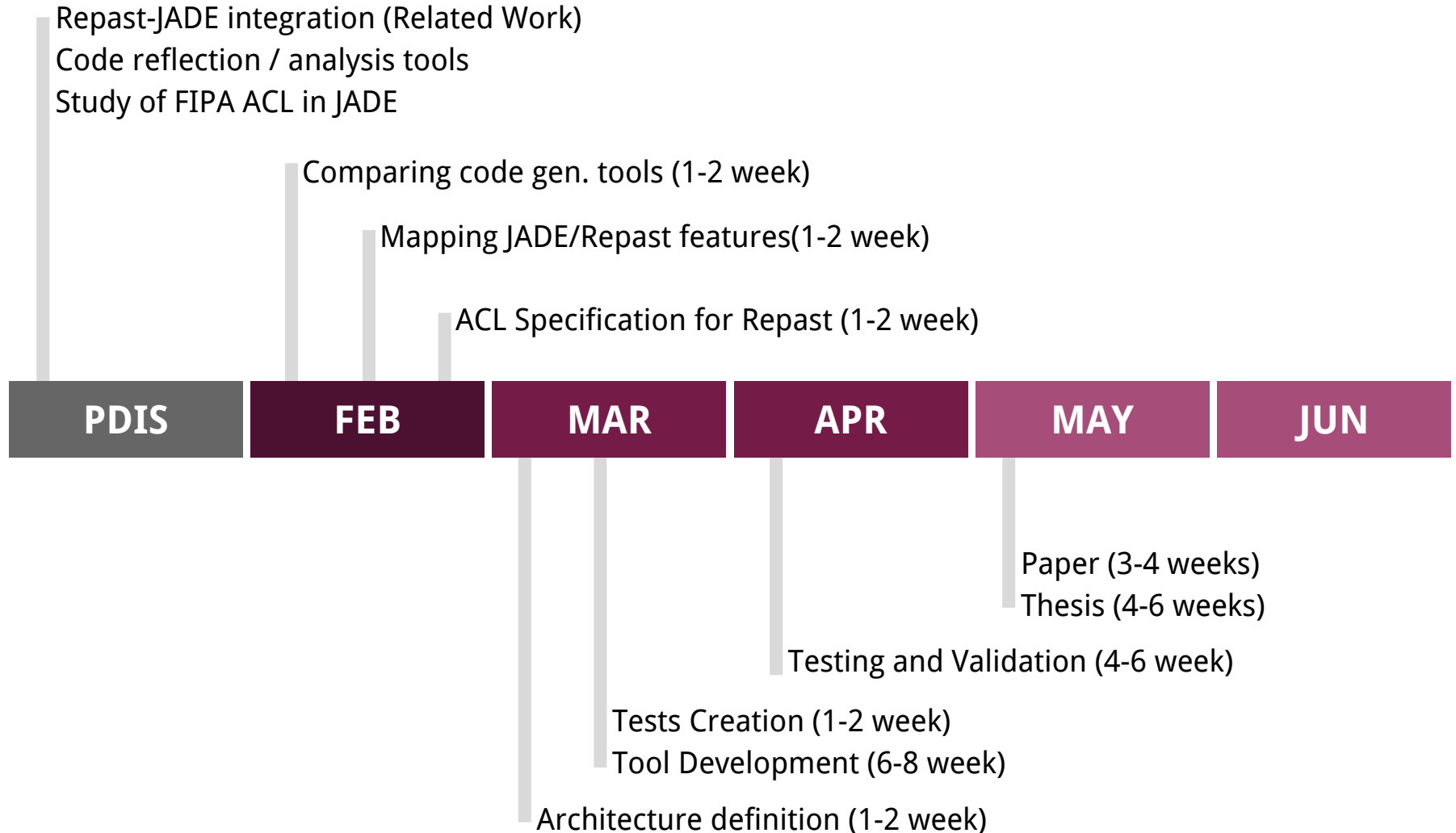
- » Java code analysis and transformation tool
- » Detects patterns, code and data structures
- » Injects and replaces code

Eclipse Java Development Tools (JDT)

- » Built-in refactoring
- » Plug-in development documentation



Workplan



Workplan

MAR

APR

Architecture definition

- » Create logical models for the code generation tool
- » Create logical models for the Repast Communications Library

Tests Creation

- » Attempt to cover all use cases of the tool (JADE and Repast features)
- » Create or select examples to test final product

Tool Development

- » Create Java library; implement FIPA protocols in Repast Symphony
- » Create tool or plug-in for code generation

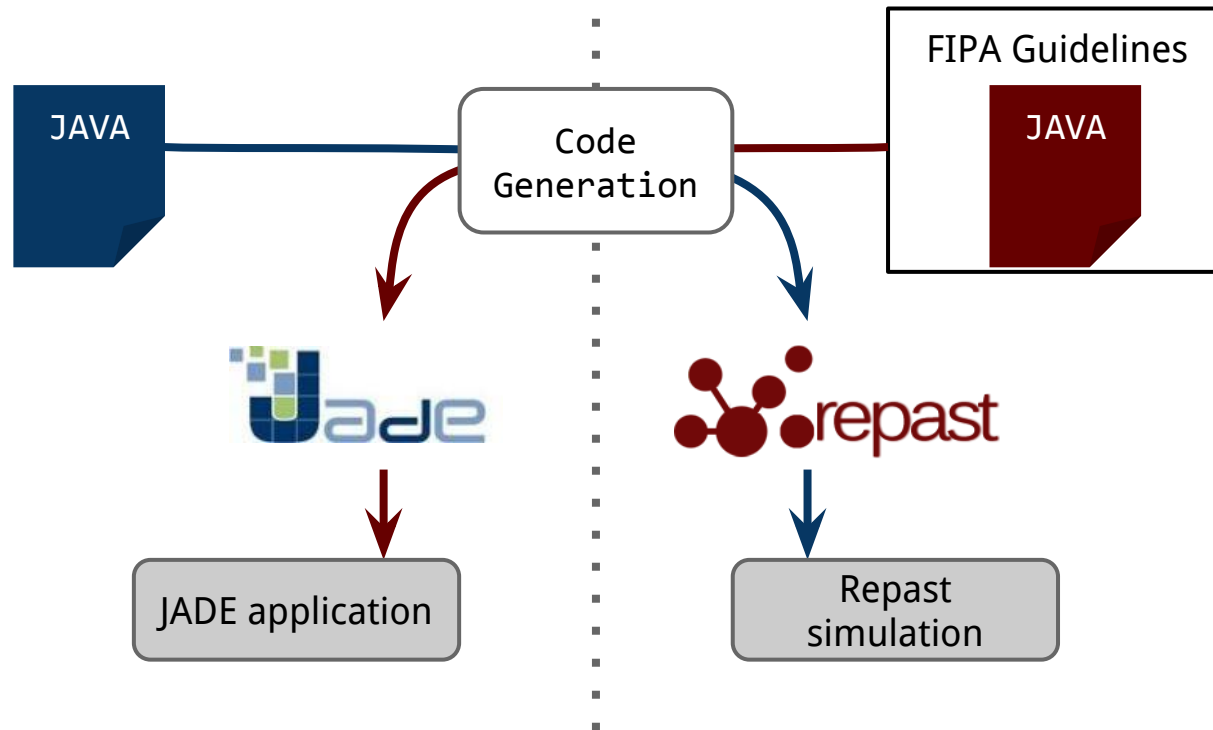
Testing and Validation

- » Test code must maintain functionality in both frameworks
- » Usability tests with Jade/Repast users

MASSim2Dev

From simulation to development in MAS

Repast-JADE automatic code generation for interaction protocols



Porto, January 27th 2014

João Pedro Lopes (lopes.joao.pedro@fe.up.pt)

Prof. Henrique Lopes Cardoso (hlc@fe.up.pt)

End