

# **Review Protocol**

# Design Patterns for Multi-Agent Systems A Systematic Literature Review



Revision

version	date
V1	2012-02-13
V2	2012-05-19
V3	2012-06-15



### Contents

Introduction	3
Purpose	3
Review Process	3
Research questions	5
Inclusion and Exclusion criteria	5
Search strategy and selection process	6
Data Collection	7
Data Analysis	8
Reliability and validity	8
Limitations	9
Significance of the study	9

### Attachments

Included Articles Excluded Articles



### Introduction

Capturing design knowledge in the form of patterns has been widely popularized is a common practice in mainstream software engineering

Design patterns allow reuse of best practices and avoiding worst. The usefulness of patterns has been proved empirically. Design patterns improve software's quality properties, like maintainability and re-usability, and speed up the development time. These factors are crucial in practice, especially for project managers, since improving them reduces costs.

During the last decade, the multi-agent system (MAS) community has put significant efforts in documenting design patterns. Despite the substantial body of work, design patterns for MAS have not received the attention they deserve, neither in the agent-oriented software community, nor among software practitioners.

As MAS have features that are widely considered as key to engineering complex distributed applications, it is important to provide a clear overview of existing patterns to make this knowledge accessible to practitioners.

The research method used in this study is a systematic literature review (SLR). An SLR is a well-defined approach to identify, evaluate and interpret all relevant studies regarding a particular research question, topic area or phenomenon of interest.

### Purpose

The study aims to provide an overview of documented design patterns for MAS. In particular, we aim to identify how the patterns are documented, whether and how the patterns are related, and for what applications the patterns have been applied. From our study, we aim to outline guidelines for future work on design patterns for MAS and in particular, their popularization in practice.

### **Review Process**

The study starts with defining an initial review protocol, followed by retrieving and selecting publications, data analysis, and report writing. We organized the harvesting of the publications in four iterations. This approach is inspired by methodologies for Agile system development as their core principles are: adaptive planning, time-boxed iterations, and rapid/flexible response to change. We deviated from a single harvesting step of a regular SLR as we wanted to learn from each iteration and adapt the search strategy accordingly.



Linnæus University

School of Computer Science, Physics and Mathematics



The review will be performed by three researchers. Two reviewers defined the initial protocol. The actual harvesting process will be performed by one reviewer, while the three reviewers will evaluate the results of the subsequent harvesting steps and adapt the search strategy in consultation. One reviewer will extract the data from the selected studies. Finally, two reviewers will synthesize and analyze the data and write the review report. These final steps will be crosschecked by the third reviewer.



### **Research** questions

Following research questions are defined:

- RQ 1: How are the patterns documented and what pattern templates are used?
- RQ 2: How are the design patterns connected?
- RQ 3: For what types of systems have the design patterns been
- applied?
- RQ 4: How can the design patterns be classified?

As the simplicity and intelligibility of a template are crucial factors in reusing patterns by practitioners, the motivation behind RQ1 is to study how pattern templates are used, find out if they require standardization and what may constitute as a common and universal vocabulary for a future pattern language. The answer of RQ2 would benefit in enhancing learnability and quicker orientation in the domain of design patterns for MAS. Visualizing associations between patterns would help in capturing a holistic picture for learners. Finding most and least MAS application fields populated with patterns (RQ3) would reveal the areas of applications where patterns need to be explored in more detail. The need of a dimensions and a roadmap when looking for specific patterns is behind including RQ4.

### Inclusion and Exclusion criteria

### **Inclusion criteria**

- IC1: papers concerning design patterns for Multi-Agent Systems
- IC2: between years: 1998-2012
- IC3: abstract and full text written in English

### **Exclusion** criteria

- EC1: paper does not include patterns described in detail or described by using a structured templates or descriptions are vague
- EC2: paper is obsolete as there are newer papers documenting the same patterns
- EC3: paper contains only review or evaluation of existing patterns for MAS



# Linnæus University

School of Computer Science, Physics and Mathematics

Motivation:

The answers to research questions are to be extracted from pattern templates and descriptions, so it is crucial that they are contained in a paper.

### Search strategy and selection process

During all four iterations a mixed search strategy was adopted in order to get more varied results. The first iteration included manual searches in the International Journal of Agent-Oriented Software Engineering (IJAOSE) and an automatic literature search based on a list of keywords in the electronic databases: ACM Digital Library, Science Direct and Lib Hub. To get a general overview of the existing contributions, following Boolean search strings were forged:

• (multi-agent OR multiagent OR MAS OR "multi-agent system" OR "multi-agent systems" OR "multiagent system" OR "multiagent systems") AND ("design pattern" OR pattern OR patterns OR "design patterns")

• (agent-based OR agent-oriented OR agent) AND ("design pattern" OR pattern OR patterns OR "design patterns")

During the second iteration, the automatic search was continued in other databases: IEEE Xplore, SpringerLink and GoogleScholar. In the third iteration, after initial selection and creation of preliminary list with patterns, other search techniques were incorporated - targeted search, ancestor search in the lists of references and descendant search aiming in finding early works and locating later articles that cite them in their references, mainly using CiteSeerX. In the fourth iteration, targeted searches were performed and other journals searched. The Knowledge Engineering Review (KER) was searched automatically, Transactions on Autonomous and Adaptive Systems (TAAS) and Journal of Autonomous Agents and Multi-Agent Systems (JAAMAS) were manually searched. The selection process was split into two phases and done concurrently and recursively during all four phases. Firstly, publications were assessed taking their title and abstract into consideration. Secondly, the full texts were screened.



## Data Collection

The collected data is stored in a SQL-based rational database and modified via a self-made MVC web application to provide more flexibility when validating.

Item id	Field	Concern
F1	Author(s)	Documentation
F2	Year	Documentation
F3	Title	Documentation
F4	Venue	Documentation
F5	Keywords	Documentation
F6	Design pattern name	RQ2
F7	Design pattern alias	RQ2
F8	Catalog pattern categories	RQ4
F9	Short pattern description	RQ3, RQ4
F10	Pattern application domains	RQ3
F11	Pattern associations	RQ2
F12	Pattern template details	RQ1

Table shows collected data items for each paper. For documentation purposes data items F1-F7 were used: author, year, title, venue, keywords and design pattern name/alias. Catalog pattern categories (F8) represent respective category to which a design pattern belongs to. The F8 item would help to determine a dimension on which a pattern resides (RQ4). Short pattern description (F9) is used to provide a brief overview of a pattern in focus on identifying pattern's dimensions and application domains. Pattern application domains item (F10) refers to specific field of application where the pattern was initially applied. Options are:

- domain-independent (not specified),
- industrial applications(process control and manufacturing, air traffic control, traffic and transportation),
- commercial applications(information management, electronic commerce, business process management),
- robotics,
- entertainment (games, interactive theater and cinema),
- simulation.

The options were elicited from papers commenting MAS practical applications as there is no explicit taxonomy of software application domains. Pattern associations item (F11) is needed to collect all related, references



patterns from template's paragraphs such as 'See also' or 'Related patterns' (RQ2). Pattern template details (F12) that consist of template text paragraphs and supplied graphical descriptions, are needed in resolving

### Data Analysis

Deduction of conclusions and recommendations for future research will be based on the synthesis. The process of synthesizing the data collection includes the following:

- 1. Listing of design patterns and articles
- 2. Analysis of data associated with design patterns
- 3. Answering research questions
- 4. Interpretation of the results

The data analysis of patterns will incorporate three different methods: meta analysis for RQ1 and RQ3, cluster analysis based on graph model for RQ2, and data classification for RQ4. Meta analysis will rely on a qualitative coding schema with dichotomous parameters. The data will be further analyzed using descriptive statistics. Graph-based model for cluster analysis aims in establishing what groups of patterns were most influential. The goal of taxonomical organization of design patterns is to create a topical classification that would help learners to find quickly searched pattern without knowing details. However, the main drawback of this method is that it relies on subjective categorization.

### Reliability and validity

Among reliability procedures will be making sure if there were no mistakes while collecting the data in the database. To ensure the validity and strengthen reliability of the research, various strategies will be followed:

- triangulation of data,
- gathering data from multiply sources,
- member checking,
- using rich and graphical descriptions to convey the findings,
- peer examination and reviewing the entire project by external auditor.



### Limitations

The limits of the study lie in the quality of results from search engines and digital databases. The selection of the papers is also limited by the quality of abstracts. Other delimitations of the study are experience of the researchers and, the time of the research.

### Significance of the study

Since the significance of the problem is dictated by the audience and benefits if the study is done, it could be claimed that that study is significant on a large scale. The significance of this study is anchored in the empirical findings, so that it would be potentially useful to every practitioner in the area of multi-agent systems.



## Attachment: Included Articles

Title	Source
Agent design patterns:	
elements of agent	http://dl.acm.org/citation.cfm?id=280784
application design	
Patterns of intelligent and	http://dl.acm.org/citation.cfm?id=280781
mobile agents	
Agent System Development	
Nethod Based on Agent	<u>http://dl.acm.org/citation.cim/id=302657</u>
Architectural Design	
Patterns for Multiagent	http://citeseerx_ist_psu_edu/viewdoc/summary?doi=10.1.1.43.9746
Coordination	http://elesteri.ist.psu.edu/viewdoc/summary/doi 10.1.1.45.2740
Patterns for Adaptive Multi-	
Agent Systems in Intelligent	http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.41.5995&rep=r
Manufacturing	ep1&type=pdf
Design Patterns for	
Multiagent Systems Design	http://www.springerlink.com/content/bc32d3lda58rfxaj
Agent Implementation	https://lirias.kuleuwan.ba/bandla/123/156780/131788
Patterns	<u>https://httas.kulcuven.oc/handle/123430763/131766</u>
Design Patterns for Self-	http://www.springerlink.com/content/8424012ai3107405/
Organising Systems	indparter www.springerinik.com/concentron/210124[5107105/
Modelling intention	http://www.dsto.defence.gov.au/publications/scientific_record.php?recor
recognition for intelligent	<u>d=3367</u>
agent systems	
Software Engineering	http://www.springerlink.com/content/27clxj83j2e9l95l/
Delegate MAS Patterns for	
Large-Scale Distributed	
Coordination and Control	http://people.cs.kuleuven.be/~danny.weyns/papers/2010EuroPlop.pdf
Applications	
Design Pattern for Self-	
Organization Multi-agent	http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6121014
Systems Based on Policy	
Agent-oriented software	
patterns for rapid and	www.elsevier.com/locate/iss
affordable robot	<u>www.uscviei.com/iocute/jss</u>
programming	
Task Knowledge Patterns	
Reuse in Multi-Agent	http://www.springerlink.com/content/77581766mn61152/
A nottern language for	
multi agent systems	https://lirias.kuleuven.be/handle/123456789/252306
Product design patterns for	
agent-hased modeling	http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6148008
Design Patterns for Mobile	
Agent-Mediated E-Business	http://ausweb.scu.edu.au/aw04/papers/refereed/nguyen/
(No) more design patterns	
for multi-agent systems	nup://di.acm.org/citation.cim/id=2095083



A system of agent-based software patterns for user modeling based on usage mining	www.elsevier.com/locate/intcom
The genesis of a pattern language for agent-based enterprise systems	http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1579162
Description and Composition of Bio-Inspired Design Patterns: the Gossip Case	http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5946189
Pattern-Based Design for Intelligent Mobile Agents	http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4430411
A Framework for Patterns in Gaia: A case-study with Organisations	http://www.springerlink.com/content/m6q7hhbc32q2dfwd/
Rapid Application Development Using Agent Itinerary Patterns	http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.26.3110&rep=r ep1&type=pdf
Agent Design Patterns Framework for MaSE/POAD Methodology	http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1618405
Security Patterns for Agent Systems	http://roar.uel.ac.uk/jspui/handle/10552/384
Design Patterns for Decentralised Coordination in Self-organising Emergent Systems	http://www.springerlink.com/content/a4957271362qu324/
Patterns as a Means for Intelligent Software Engineering	http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.41.2195
A Set of Agent Patterns for a More Expressive Approach	http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.22.7623
Reusable Patterns for Agent Coordination	http://www.springer.com/computer/communication+networks/book/978- 3-540-41613-5
Role agent pattern: a developer guideline	http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1245631
Social Patterns for Designing Multiagent Systems	http://www.ucllouvain.be/cps/ucl/doc/iag/documents/WP_88_Do_Kolp_ Pirotte.pdf
The Reflective Blackboard Architectural Pattern	http://www.springerlink.com/content/0r1f2n9ax15j3n5v/
A Pattern Language for Motivating the Use of Agents	http://www.springerlink.com/content/w8jca26a380ul5uk/
Design Patterns for Agent- Based Service Composition in the Web	http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1579167
Multi-Agent Architectures as Organizational Structures	http://www.springerlink.com.proxy.lnu.se/content/t00618414v564832



Describing Agent Based		
Real-Time Distributed	http://dl.acm.org/citation.cfm?id=1233341.1233370	
Systems using Design		
Patterns		
Patterns for Modelling	http://www.apringarlink.com.prov.lpu.go/content/r502h451024771m0	
Agent Systems with Tropos	<u>http://www.springerlink.com.proxy.inu.se/content/r502n451034771m9</u>	
Towards Pattern-Oriented		
Design of Agent-based	http://www.springerlink.com/content/6p55104x43391327/	
Simulation Models		



# Attachment: Excluded Articles

Title	Exclusion Criteria
Agents in Object Oriented Software Engineering	EC1
Agents in Object-Oriented Software Engineering	LUI
Pattern-Oriented Design for Multi-Agent	EC1
System: A Conceptual Model	
A Semantic Description For Agent Design Patterns	EC1, EC2, EC3
A performance model for task and interaction patterns in mobile agent	EC3
systems	
Advanced separation of concerns in agent-oriented	EC1, EC3
An agent design pattern classification scheme capturing the notions of	EC3
agency in agent design patterns	
Application of Patterns in Agent System Design	EC3, EC1
A comprehensive view of agent-oriented patterns	EC3
A multi-agent approach to unit commitment problems	EC3
A Pattern Language for FIPA Agent Interface	EC1, EC3
A Pattern Template for Intelligent Agent Systems	EC3
A Pattern-Based Coordination and Test Framework for Multi-Agent	EC3, EC1
Simulation of Production Automation Systems	
A pattern-based development tool for mobile agents	EC3,EC1
A pattern-oriented design agent	EC1
A Petri net model of the meeting design pattern for mobile-stationary	EC3
agent interaction	
A possible approach to the development of robotic multi-agent systems	EC2
Agent Oriented Design Patterns	EC2, EC1
Agent-oriented architecture for intelligent service robot	EC1
An approach to modelling and applying mobile agent design patterns	EC2
An Architectural Strategy for Self-Adapting Systems	EC1
An architecture for reusable development in distributed system	EC1
An architecture-centric approach for multi-agent system development	EC1
and application	
Analysis of Typical Patterns of the Agent Construction Institution in	EC1, EC3
Public Project	
Applying the MVC Design Pattern to Multi-Agent Systems	EC3
Architectures for novel energy infrastructures Multi-agent based	EC1
coordination patterns	
Integrating Patterns and Agent-Oriented Methodologies to Provide	EC1
Better	
Solutions for the Development of Secure Agent-Based Systems	
CoDesign - a collaborative pattern design system based on agent	EC1
A Catalogue of Decentralised Coordination Mechanisms for Designing	EC2
Self-Organising Emergent Applications	
Design of agent-oriented pattern templates	EC3
Design Patterns to Enable Agent-based Mobile Service Application	EC1
Development	
Improved template for agent pattern description	EC3
Introducing Pattern Reuse in the Design of Multi-Agent Systems	EC2
Mobile agent interoperability patterns and practice	EC3
Patterns and Protocols for Agent-Oriented Software Development	EC3
Pattern Repository and Reuse in the PASSI Methodology	EC2



Tools and patterns in designing multi-agent systems with PASSI	EC2
Using Tropos to model agent based architectures for adaptive systems a	EC2
case study in ambient intelligence	
The use of design patterns for the development of multi-agent systems	E3,E1
Patterns for e-Commerce Agent Architectures: Using Agents as	E2
Delegates	
A Goal-Based Organizational Perspective on Multi-Agent Architectures	EC3
A Social Organization Perspective on Software Architectures	EC2
MAS Development Reusing through Agent Oriented Design Patterns	EC2
MAS Development Reusing through Agent Oriented Design Patterns	EC2
Facilitator Agent Design Pattern of Procurement Business Systems	EC3
The agent pattern for mobile agent systems	EC2