ERC Advanced Grant 2017
Research proposal [Part B1]

White-Box Self-Programming Mechanisms

WhiteMech

Cover page:

– Principal investigator (PI): **Giuseppe De Giacomo**
– Host institution: **Università degli Studi di Roma "La Sapienza"**
– Proposal duration: **60 months**

We are witnessing an increasing availability of **mechanisms** that offer some form of programmability. These include software, manufacturing devices, smart objects and spaces, intelligent robots, business process management systems, and many others. All these mechanisms are currently being revolutionized by means of advances in Machine Learning, which facilitate the availability of sensing (vision and language understanding) and actuation capabilities (robot-arm movement and camera pointing). However, such mechanisms are held back by the fact that the connections between components are still organized and coordinated through standard programming.

WhiteMech aims at developing the science and the tools for a new generation of mechanisms to emerge: mechanisms that are able to **program themselves** without human intervention, automatically tailor their behavior so as to achieve desired goals, maintain themselves within safe boundaries in a changing environment, and follow regulations and conventions that evolve over time. Crucially, empowering mechanisms with self-programming carries significant risks and we must be able to **balance power with safety**. For this reason WhiteMech intends to realize mechanisms that are **white-box**, that is, that are guarded by human guided specifications, which, together with the synthesized behavior, are fully verifiable and comprehensible in human terms.

Remarkable recent discoveries by the applicant in **Reasoning about Action** and **Generalized Planning** in AI, and their connections to Verification and Synthesis in Formal Methods, and Data-aware Processes in Databases, chart an unanticipated novel path to produce a breakthrough in realizing **powerful self-programming mechanisms**, while keeping them **human-comprehensible and safe by design**.

WhiteMech will ground its scientific results upon diverse **driving applications**, including smart manufacturing (Industry 4.0), smart spaces (IoT) and business process management systems (BPM).

a Extended Synopsis of the scientific proposal

LONG TERM VISION

Consider the following scenario.

> *After a long week-end, the human supervisor inspects the manufacturing system and notices that a production line has slowed down significantly, though it is still producing. She queries the system on what it is doing. The system reveals to her the revised process, which is avoiding the use of the production island 176-176, by repurposing the tools in island 176-671 and sending items there. She then queries why the system has reprogrammed itself to do so. The system answers by showing that on Sunday 11:43pm the island 176-176 started to produce an unacceptable percentage of defective items, based on tests performed during production. So, the system restructured the process to achieve the specified objectives (quality and throughput) as well as it could in the presence of the faulty island, instead of shutting down the production line: the system analyzed the available capabilities and reprogrammed itself resulting in the current revised process.*

In the scenario above we have a **mechanism** (the manufacturing system) with multiple components, possibly using Machine Learning (ML) to provide sensing and acting capabilities (detecting defective items, reconfiguring tools parameters), which can be suitably organized to enact a *dynamic behaviour* (the production process) to meet its *specifications* (constraints on the product and production model). The mechanism has the ability to monitor and detect faulty parts of the process, and crucially it has **self-programming** abilities that it can use to modify its current behavior *without human intervention*. Notably, the mechanism can be *queried* to display, in terms understandable to humans, under which circumstances and for fulfilling which *specifications*, it reprogrammed itself. Moreover it can be queried on whether its *self-synthesized program* meets any dynamic property of interest to the human supervisor (e.g., for "what-if" analysis). In a slogan, the mechanism is **white-box**.

The overarching objective of WhiteMech is to make this vision a reality:

> WhiteMech *aims at laying the theoretical foundations and developing practical methodologies of a science and engineering of* ***white-box self-programming mechanisms****.*

To make apparent the significance of this enterprise, WhiteMech will ground its research in three driving applications considered of pivotal importance in the current socio-economic context, namely:

1. **Smart Manufacturing**, where significant research efforts are focusing on improving flexibility, agility and productivity of manufacturing systems, under the umbrella term *Industry 4.0*, or *4th industrial revolution*.[1]
2. **Internet of Things**, which is rising as a virtual fabric that connects "things" equipped with chips, sensors and actuators and allows for building **smart objects** and **smart spaces** with high levels of awareness of the environment and its human occupants.[2]
3. **Business Process Management**, which advocates explicit conceptual descriptions of a process to be enacted within an organization or possibly across organizations, and which is instrumental to business processes improvement, the top business strategy of CIOs in organizations according to Gartner. [3]

Interestingly forms of self-programmability have been advocated in all the above contexts. For example, it is advocated that cyber-physical systems in Manufacturing or Internet of Things should be able to **adapt** themselves to current users and environment by **exploiting information gathered at run-time**. However it is considered **impossible to determine a priori all possible adaptations** that may be needed; thus self-programming abilities would be highly desirable [87]. In Business Processes, it is considered important for the next generation of process management systems to allow processes to automatically **recover** when unanticipated exceptions occur, without explicitly defining a priori

[1] M. Lorenz et al. *Man and Machine in Industry 4.0: How Will Technology Transform the Industrial Workforce Through 2025?* The Boston Consulting Group. 2015.

[2] C. MacGillivray et al. *Worldwide Internet of Things Forecast Update, 2016-2020.* IDC. Doc # US40755516. 2016.

[3] Gartner Group. *BPM Survey Insights.* Gartner Report. http://www.gartner.com/it/page.jsp?id=1740414.

recovery policies, and **without the intervention of domain experts** at runtime. These self-programming abilities would reduce costly and error-prone manual ad-hoc changes, and would relieve software engineers from mundane adaptation tasks [73]. Note that some of these concerns have been shared by *autonomic computing*, which has promoted self-configuration, self-healing, self-optimization, and self-protection, though by using policies provided by IT professionals [67]. Sophisticated languages and methodologies for streamlining the development of adaptation and exception handling recovery procedures have been developed, however IT professionals still need to write all code by hand in the end [26, 74].

Although the interest is clearly apparent, **currently these self-programing abilities are missing** in actual mechanisms, and science is focusing on limited forms of self-programming, e.g., for exception handling and recovery, or forms of composition and autonomic reconfiguration to be applied at design time [62].

WhiteMech intends to **make a quantum leap in mechanisms' self-programming abilities**. Through enhanced self-programming abilities such mechanisms can, e.g.:

– *Achieve desired goals*, that is guarantee that a certain desired state of affairs is eventually reached. In the above example a manufacturing system automatically reconfigures the fabrication process if some workstation is producing too many defective items, by changing the sequencing of processing units so as to temporarily cut-out the defective tool from the process, so as to achieve an acceptable error rate.
– *Maintain themselves within a safe boundary* against unanticipated changes in the environment in which they operate. For example a smart space system may keep the desired temperature and humidity in a museum room at some desired level, even in presence of an unforeseen large crowd of visitors, possibly by momentarily repurposing other actuators, such as a secondary air conditioning systems typically used only in case of failure of the main one.
– *Keep following regulations and conventions* that evolve over time while enacting their behavior. For example, to answer a new privacy regulation, a business process may refine its behavior to guarantee that the sensitive data are erased from the system before the completion of each process instance.

More generally, WhiteMech wants to enable mechanisms to act in an informed and intelligent way in their environment, by changing the way they behave as a consequence of the information they acquire from the external world and exchange with the humans operating therein.

Since "*with great power comes great responsibility*", introducing advanced forms of self-programming calls for the ability to make the behavior automatically synthesized by the mechanism **comprehensible** to human supervisors, who are thus able to control and guide it. So it is indeed crucial to develop self-programming mechanisms that **can be queried**, i.e., that are **white-box**: in every moment the mechanism can be queried for the status of its specifications, and on whether its behavior meets any dynamic property of interest to the human supervisors. Ultimately it is the fact that the resulting behavior is **comprehensible in human terms** that will make white-box self-programming mechanisms **trustworthy** [25, 76].

In the first example above, both the reconfiguration goal (cutting out a defective tool) and how the fabrication process has been modified need to be explicitly understandable by the humans analyzing the manufacturing system. In the second example, the sudden repurposing of the secondary air conditioning system also needs to be understandable to humans as a reaction to avoid violating certain safety requirements. Similarly, in the third example, the goal of erasing sensitive data from the system, and even more importantly how this is achieved, must be understandable.

We stress that the need to move towards **white-box** approaches is advocated by a large part of the **AI community** [82] as well as the **CS community** [1], and has been recently taken up by DARPA within the context of machine learning, through the DARPA-BAA-16-53 "Explainable Artificial Intelligence (XAI)" program.[4] *Knowledge representation, the primary field of the PI, will be central for realizing the shift towards a white-box approach.*

We observe that there is a well justified enthusiasm for using ML-techniques to develop smarter

[4] http://www.darpa.mil/program/explainable-artificial-intelligence

systems. While typically these ML-components are black-boxes, in the sense that how they work remains opaque to humans [75, 90], there is currently much work ongoing on incorporating some forms of human-control into such ML-systems to provide safety guarantees [2, 93, 61]. WhiteMech intends to fully support the integration of such ML-components with suitable safety guarantees into white-box self-programming mechanisms.

OBJECTIVES

Towards the goal of building **white-box self-programming mechanisms**, WhiteMech will address 5 specific objectives, described below together with the area involved in achieving them.

1. **Equip mechanisms with general self-programming abilities.** Mechanisms need general self-programming abilities, not restricted to a particular task, such as exception recovery, but ready to refine and modify the behavior of the mechanisms as new opportunities or constraints arise. Self-programming abilities are needed **while mechanisms are in operation**, that is while the mechanisms are executing, not at design time. WhiteMech aims at advanced forms of **process synthesis** as those studied in **Verification and Synthesis** in Formal Methods [78, 52] and especially in **Generalized Planning** in AI [94, 47, 48, 24].

2. **Make self-programming mechanisms comprehensible and verifiable by humans.** WhiteMech requires specifications, solutions (synthesized programs) and the relationship between them to be **comprehensible to humans**. This means that specifications and solution spaces must be **semantically** described at high-level using predicates that are understandable to humans, as advocated by **Knowledge Representation** (KR) [7, 53, 54, 17, 89, 70, 71]. Clearly, a crucial step towards the understandability, is to be able to **verify** replanned behaviors against their specifications as advocated by **Verification and Synthesis**.

3. **Make self-programming mechanisms data-aware.** During the execution, new facts about the world are observed, learned, or received as input. This calls for a representation that distinguishes **intensional information** such as that provided by knowledge of the domain, from **extensional information** provided by actual data. Self-programming mechanisms leverage on the intensional information to be able to interpret new data (extensional information) acquired, observed, and learned. Notice that this calls for a **relational (first-order) representation of the state**. New results on verifiability of **Data-Aware Processes**, based on faithful abstraction fo finite-state transition systems, are crucial for this [27, 5, 28, 66, 9, 36, 21].

4. **Support component-based approaches.** By no means should we consider programmable mechanisms to be composed of a single unit. Indeed, understandability calls for building **high-level components** that are relatively simple to understand, verify and combine. Then, it is crucial to study how **composing** "correct" components leads to an overall "correct" behavior. WhiteMech will leverage work on composition and customization in Service Oriented Computing and more recently in **Reasoning about Action** in KR and **Generalized Planning** in AI [15, 91, 12, 45, 33], also looking at execution **monitoring** to detect failure, identify responsible components, and synthesize recovery [29, 73].

5. **Allow learning and stochastic decisions, while remaining within safe bounds.** We want to allow mechanisms to have forms of decision making that resist formal analysis (at least in human terms), because we want to make use of the possibilities that advancements brought about by deep learning, MDP's, and reinforcement learning. However, while the actual execution could be chosen stochastically, we do want to have safety guarantees on **all possible generated executions** [2, 93, 61]. In this way, it is the **entire space of solutions** that has **formal guarantees**, and the specific solution chosen by the learning algorithm or the stochastic decision maker will also satisfy them. We will study this objective in the context of **Planning under stochastic uncertainty** [59, 60].

Recent foundational results by the PI chart a novel path that within WhiteMech *will revolutionize **Reasoning about Action** in KR and **Generalized Planning** in AI by introducing rich objectives, data, and componentization in order to produce a **breakthrough in engineering self-programming mechanisms that are human-comprehensible and safe by design.***

METHODOLOGY

The scientific work in WhiteMech will be methodologically structured into 3 broad research streams:

- **Principles and Foundations** that will deal with the scientific foundations of white-box self-programming mechanisms.
- **Algorithms and Tools** that will deal with the development of practical algorithms, optimizations and tools for realizing such mechanisms.
- **Applications and Evaluation** that will evaluate white-box self programming mechanisms in the three business critical driving applications mentioned above.

The **Principles and Foundations** and **Algorithms and Tools** streams will cut across 5 workpackages (WPs) roughly corresponding to the 5 objectives above. The **Applications and Evaluation** stream will be further refined into 3 separate WPs for the 3 driving applications. WPs are described in Part B2.

Below we sketch WhiteMech's scientific approach, focusing on **novelty** and **feasibility**.

WhiteMech will take from **Planning** in AI [59] (which in turn coming from **Reasoning about Action** in KR [81]) the idea of having **human-comprehensible** descriptions of the domain (the mechanism and its capabilities) and of the goals (the task specifications). However instead of considering simple domain specifications, as e.g., in PDDL, WhiteMech will look at extensions that use **rich semantic descriptions** from Knowledge Representation and **componentization** from behavior compositions in Reasoning about Action, to which the PI has contributed significantly in the years [46, 34, 41, 85, 84, 40, 35, 38, 39, 6]. Particular attention will be given to computational effectiveness, in line with some recent exploratory work by the PI [36, 37, 21]. Notice that on the other hand WhiteMech does not aim at defining new concrete representation languages. Instead it intends to use well-known formalisms such as BPMN, UML, OWL, etc. as concrete languages, though with a precise formal semantics to allow for automated reasoning, verification and synthesis, see e.g., [10, 44].

Another crucial difference wrt Planning is that, instead of considering simple planning tasks, WhiteMech will make use of the mathematically elegant theory of Reactive Synthesis [78] developed in **Verification and Synthesis** in the last 30 years [52], which however has not found widespread practical application because of the **intrinsic difficulties** of certain algorithms and constructions [95, 58]. We aim at **sidestepping these difficulties all-together**, by focusing on non-traditional kinds of specification formalisms proposed recently in, e.g., Reasoning about Action and Generalized Planning, such as LTL and LDL on finite traces, recently proposed by the PI together with Moshe Vardi (Rice U, Huston) [50, 47, 48] and adopted in generalized forms of Planning in AI [94, 24] and in declarative business processes in BPM [96, 29, 42], as well as safe/co-safe LTL/LDL formulas, which have been shown to be more expressive than expected while remaning **well-behaved** [57, 56, 69, 55]. Solvers for these are substantially simpler that for general reactive synthesis, being based on reachability and safety games, which are amenable to efficient implementations.

Moreover, like in Planning, but differently form Reactive Synthesis, we expect mechanisms to be able to handle quickly and efficiently most cases, i.e., those cases that do not require to solve difficult, "puzzle-like", situations. Indeed while the Planning community has concentrated on simpler forms of process synthesis (reachability of a state of affairs), it has developed a sort of **science of search algorithms for Planning**, which has brought about exceptional scalability improvements (orders of magnitude) in the last decade [80, 92, 72, 42]. WhiteMech intends to devise new algorithms for solving reachability and safety games that are based on heuristic search as adopted in Planning, but also considering symbolic techniques adopted in synthesis by model checking [13]. The PI has been pioneering cross-fertilization of Planning and Synthesis since the end of the '90 [49, 23, 86, 32, 77, 42]. More recently the PI has established **tight connections between synthesis and generalized forms Planning** [64, 65, 43, 14] as well as between Planning and Behavior Composition [83, 45, 33, 19].

Crucially, differently from current work in Planning and in Verification and Synthesis, which operate with a propositional representation of the state, WhiteMech does not want to discard data, and hence it will need to consider a first-order or relational representation of the state. This gives rise to infinite transition systems which are generally problematic to analyze. However the PI has already shown within the EU FP7-ICT-257593 ACSI: Artifact-Centric Service Interoperation, that such difficulties can be overcome in notable cases [11, 18, 5, 20] in the context of **Data-Aware Processes** in Databases. A key decovery is that under natural assumptions these infinite-state transitions systems admit faithful **abstractions** to finite-state ones, hence enabling the possibility of using the large body of techniques developed within Verification and Synthesis in Formal Methods. Moreover important advancements in undestanding how to deal with such complexity have been established as well as relationships with Reasoning about Action in KR [63, 9, 22, 21, 6]. We will leverage on these ideas to lift our results so as to handle data.

WHITEMECH *will, in addition, be the spark that will bring together and cross-fertilize four distinct research areas, namely* ***Reasoning about Action*** *in* Knowledge Representation*,* ***Data-aware Processes*** *in* Databases*,* ***Verification and Synthesis*** *in* Formal Methods*, and* ***Generalized Planning*** *in* AI*.*

The PI has profoundly contributed to all these areas and is in a unique position to lead this cross-fertilization.

To foster such a cross-fertilization WHITEMECH will establish research exchanges with several research groups in the above mentioned areas.

- **Reasoning about Action.** Collaborations with Yves Lesperance (York U., Toronto, Canada), Hector Levesque (U. Toronto, Canada), Sebastian Sardina (RMIT, Melbourne, Australia), and Yongmei Liu (Sun Yat-sen U., Guangzhou, China).
- **Data-Aware Processes.** Collaborations with Rick Hull (IBM Research, USA), Jianwen Su (UCSB, USA), Diego Calvanese and Marco Montali (U. Bolzano, Italy), and with Alessio Lomuscio (Imperial College, London, UK).
- **Verification and Synthesis.** Collaborations with Moshe Vardi (Rice U.) on automata-based verification and synthesis, Sasha Rubin (U. Napoli, Italy), and Benjamin Aminof (TU Wien, Austria) on game-based verification and synthesis, and with Nello Murano (U. Napoli, Italy) on devising parallel algorithms to run on multicore GPUs [3, 4].
- **Generalized Planning.** Collaborations with Hector Geffner (UPF, Barcelona), Blai Bonet (U. Simon Bolivar, Caracas) and Alfonso Gerevini (U. Brescia, Italy), and Malte Helmert (U. Basil).

Moreover the PI will collaborate with Ronen Brafman (Ben-Gurion U.) to explore how to **incorporate ML-components** into mechanisms, and study Planning and Synthesis for temporally-extended goals in **non-Markovian** MDPs and reinforcement learning (first ideas in [8, 69, 16]).

Application and Evaluation WHITEMECH will ground its scientific results in three diverse real **application** to demonstrate the actual utilization of the scientific achievements within the project: Smart manufacturing (Industry 4.0), Smart spaces (IoT), and Business Processes Management Systems (BPM). The PI and his group at Sapienza have contributed to all these fields, see e.g., [30, 31, 51]. Moreover the PI has applied advanced science to real-cases in the area Semantic Data Integration where he and his group have invented the Ontology-Based Data Access paradigm, possibly the most successful approach for Semantic Data Integration [79, 88, 68]. Such an approach has matured to the point that the PI founded a Sapienza Start-Up **OBDA Systems** (http://www.obdasystems.com) to commercially exploit it in real data integration scenarios.

HIGH RISK, HIGH GAIN

WHITEMECH boldly aims at **bringing together and cross-fertilizing the above cited four distinct research areas** with overlapping interests but developed by different communities with different view-points, to produce a **breakthrough** that will make white-box self programming mechanisms a reality. The **high risk** of this enterprise is **mitigated** by the PI expertise, who (by cross-fertilize some of these areas) has already obtained initial foundational results, such as effective techniques for verification of data-aware processes in spite of them being infinite state in nature due to data [5, 36, 21], and feasibility results for synthesis, which sidestep some intrinsic difficulties of reactive synthesis algorithms and constructions (e.g., determinization) [50, 47, 48]. WHITEMECH will act as a **catalyst** for these areas and, together with current ML advancements, to bring about a **novel AI framework for self-programmability** that puts **human-comprehensibility at the center of the stage**.

References

[1] ACM U.S. Public Policy Council and ACM Europe Policy Committee. Statement on algorithmic transparency and accountability. ACM, 2017.

[2] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané. Concrete problems in AI safety. *CoRR*, abs/1606.06565, 2016.

[3] R. Arcucci, U. Marotta, A. Murano, and L. Sorrentino. Parallel parity games: a multicore attractor for the zielonka recursive algorithm. In *ICCS*, pages 525–534, 2017.

[4] R. Arcucci, A. Murano, G. Perelli, and L. Sorrentino. A parallel model for reachability games. In *Submitted*, 2017.

[5] B. Bagheri Hariri, D. Calvanese, G. De Giacomo, A. Deutsch, and M. Montali. Verification of relational data-centric dynamic systems with external services. In *PODS*, pages 163–174, 2013.

[6] B. Banihashemi, G. De Giacomo, and Y. Lespérance. Abstraction in situation calculus action theories. In *AAAI*, pages 1048–1055, 2017.

[7] C. Baral. *Knowledge Representation, Reasoning and Declarative Problem Solving.* Cambridge Univ. Press, 2010.

[8] D. Beck and G. Lakemeyer. Reinforcement learning for Golog programs with first-order state-abstraction. *Logic Journal of the IGPL*, 20(5):909–942, 2012.

[9] F. Belardinelli, A. Lomuscio, and F. Patrizi. Verification of agent-based artifact systems. *J. Artif. Intell. Res. (JAIR)*, 51:333–376, 2014.

[10] D. Berardi, D. Calvanese, and G. De Giacomo. Reasoning on UML class diagrams. *Artif. Intell.*, 168(1-2):70–118, 2005.

[11] D. Berardi, D. Calvanese, G. De Giacomo, R. Hull, and M. Mecella. Automatic composition of transition-based semantic web services with messaging. In *VLDB*, pages 613–624, 2005.

[12] P. Bertoli, M. Pistore, and P. Traverso. Automated composition of web services via planning in asynchronous domains. *Artif. Intell.*, 174(3-4):316–361, 2010.

[13] R. Bloem, B. Jobstmann, N. Piterman, A. Pnueli, and Y. Sa'ar. Synthesis of reactive(1) designs. *J. Comput. Syst. Sci.*, 78(3), 2012.

[14] B. Bonet, G. De Giacomo, H. Geffner, and S. Rubin. Generalized planning: Non-deterministic abstractions and trajectory constraints. In *IJCAI*, 2017.

[15] A. Bouguettaya, Q. Z. Sheng, and F. Daniel, editors. *Web Services Foundations.* Springer, 2014.

[16] R. Brafman, G. De Giacomo, M. Mecella, and S. Sardina. Service composition under probabilistic requirements. In *GenPlan Workshop at ICAPS*, 2017.

[17] G. Brewka, S. Ellmauthaler, and J. Pührer. Multi-context systems for reactive reasoning in dynamic environments. In *ECAI*, pages 159–164, 2014.

[18] D. Calvanese, G. De Giacomo, R. Hull, and J. Su. Artifact-centric workflow dominance. In *ICSOC*, pages 130–143, 2009.

[19] D. Calvanese, G. De Giacomo, M. Lenzerini, and M. Vardi. Regular open APIs. In *KR*, pages 329–338, 2016.

[20] D. Calvanese, G. De Giacomo, M. Montali, and F. Patrizi. Verification and synthesis in description logic based dynamic systems. In *RR*, pages 50–64, 2013. Best paper award.

[21] D. Calvanese, G. De Giacomo, M. Montali, and F. Patrizi. First-order μ-calculus over generic transition systems and applications to the situation calculus. *Information & Computation*, 2017. To appear.

[22] D. Calvanese, G. De Giacomo, and M. Soutchanski. On the undecidability of the situation calculus extended with description logic ontologies. In *IJCAI*, pages 2840–2846, 2015.

[23] D. Calvanese, G. De Giacomo, and M. Y. Vardi. Reasoning about actions and planning in LTL action theories. In *KR*, pages 593–602, 2002.

[24] A. Camacho, E. Triantafillou, C. J. Muise, J. A. Baier, and S. A. McIlraith. Non-deterministic planning with temporally extended goals: LTL over finite and infinite traces. In *AAAI*, pages 3716–3724, 2017.

[25] C. Castelfranchi and R. Falcone. *Trust Theory: A Socio-Cognitive and Computational Model*. Wiley Series in Agent Technology. John Wiley & Sons, 2010.

[26] V. G. Cerf. A long way to have come and still to go. *Commun. ACM*, 58(1):7, 2015.

[27] J. Claßen and G. Lakemeyer. A logic for non-terminating Golog programs. In *KR*, pages 589–599, 2008.

[28] J. Claßen, M. Liebenberg, G. Lakemeyer, and B. Zarrieß. Exploring the boundaries of decidable verification of non-terminating Golog programs. In *AAAI*, pages 1012–1019, 2014.

[29] G. De Giacomo, R. De Masellis, M. Grasso, F. M. Maggi, and M. Montali. Monitoring business metaconstraints based on LTL and LDL for finite traces. In *BPM*, pages 1–17, 2014.

[30] G. De Giacomo, C. Di Ciccio, P. Felli, Y. Hu, and M. Mecella. Goal-based composition of stateful services for smart homes. In *OTM*, pages 194–211, 2012.

[31] G. De Giacomo, M. Dumas, F. M. Maggi, and M. Montali. Declarative process modeling in BPMN. In *Advanced Information Systems Engineering - 27th International Conference, CAiSE 2015, Stockholm, Sweden, June 8-12, 2015, Proceedings*, pages 84–100, 2015.

[32] G. De Giacomo, P. Felli, F. Patrizi, and S. Sardiña. Two-player game structures for generalized planning and agent composition. In *AAAI*, 2010.

[33] G. De Giacomo, A. E. Gerevini, F. Patrizi, A. Saetti, and S. Sardiña. Agent planning programs. *Artif. Intell.*, 231:64–106, 2016.

[34] G. De Giacomo, Y. Lespérance, and H. J. Levesque. Congolog, a concurrent programming language based on the situation calculus. *Artif. Intell.*, 121(1-2):109–169, 2000.

[35] G. De Giacomo, Y. Lespérance, and C. J. Muise. On supervising agents in situation-determined congolog. In *AAMAS*, pages 1031–1038, 2012.

[36] G. De Giacomo, Y. Lespérance, and F. Patrizi. Bounded situation calculus action theories. *Artif. Intell.*, 237:172–203, 2016.

[37] G. De Giacomo, Y. Lespérance, F. Patrizi, and S. Sardiña. Verifying ConGolog programs on bounded situation calculus theories. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, pages 950–956, 2016.

[38] G. De Giacomo, Y. Lespérance, F. Patrizi, and S. Vassos. LTL verification of online executions with sensing in bounded situation calculus. In *ECAI*, pages 369–374, 2014.

[39] G. De Giacomo, Y. Lespérance, F. Patrizi, and S. Vassos. Progression and verification of situation calculus agents with bounded beliefs. *Studia Logica*, 104(4):705–739, 2016.

[40] G. De Giacomo, Y. Lespérance, and A. R. Pearce. Situation calculus based programs for representing and reasoning about game structures. In *KR*, 2010.

[41] G. De Giacomo, H. J. Levesque, and S. Sardiña. Incremental execution of guarded theories. *ACM Trans. Comput. Log.*, 2(4):495–525, 2001.

[42] G. De Giacomo, F. M. Maggi, A. Marrella, and F. Patrizi. On the disruptive effectiveness of automated planning for LTL_f-based trace alignment. In *AAAI*, pages 3555–3561, 2017.

[43] G. De Giacomo, A. Murano, S. Rubin, and A. D. Stasio. Imperfect-information games and generalized planning. In *IJCAI*, pages 1037–1043, 2016.

[44] G. De Giacomo, X. Oriol, M. Estañol, and E. Teniente. Linking data and BPMN processes to achieve executable models. In *CAISE*, pages 612–628, 2017.

[45] G. De Giacomo, F. Patrizi, and S. Sardiña. Automatic behavior composition synthesis. *Artif. Intell.*, 196:106–142, 2013.

[46] G. De Giacomo, R. Reiter, and M. Soutchanski. Execution monitoring of high-level robot programs. In *KR*, pages 453–465, 1998.

[47] G. De Giacomo and M. Vardi. Synthesis for LTL and LDL on finite traces. In *IJCAI*, pages 1558–1564, 2015.

[48] G. De Giacomo and M. Vardi. LTL_f and LDL_f synthesis under partial observability. In *IJCAI*, pages 1044–1050, 2016.

[49] G. De Giacomo and M. Y. Vardi. Automata-theoretic approach to planning for temporally extended goals. In *ECP*, pages 226–238, 1999.

[50] G. De Giacomo and M. Y. Vardi. Linear temporal logic and linear dynamic logic on finite traces. In *IJCAI*, pages 854–860, 2013.

[51] L. de Silva, P. Felli, J. C. Chaplin, B. Logan, D. Sanderson, and S. M. Ratchev. Synthesising industry-standard manufacturing process controllers. In *AAMAS*, pages 1811–1813, 2017.

[52] R. Ehlers, S. Lafortune, S. Tripakis, and M. Y. Vardi. Supervisory control and reactive synthesis: a comparative introduction. *Discrete Event Dynamic Systems*, 27(2):209–260, 2017.

[53] T. Eiter, E. Erdem, M. Fink, and J. Senko. Updating action domain descriptions. *Artif. Intell.*, 174(15):1172–1221, 2010.

[54] T. Eiter, G. Gottlob, and T. Schwentick. The model checking problem for prefix classes of second-order logic: A survey. In *Fields of Logic and Computation, Essays Dedicated to Yuri Gurevich on the Occasion of His 70th Birthday*, pages 227–250, 2010.

[55] P. Faymonville, B. Finkbeiner, M. N. Rabe, and L. Tentrup. Encodings of bounded synthesis. In *TACAS*, pages 354–370, 2017. Supported by the European Research Council (ERC) Grant OSARES (No. 683300).

[56] E. Filiot, N. Jin, and J. Raskin. Antichains and compositional algorithms for LTL synthesis. *Formal Methods in System Design*, 39(3):261–296, 2011.

[57] B. Finkbeiner and S. Schewe. Bounded synthesis. *STTT*, 15(5-6):519–539, 2013.

[58] S. Fogarty, O. Kupferman, M. Vardi, and T. Wilke. Profile trees for Büchi word automata, with application to determinization. In *GandALF*, 2013.

[59] H. Geffner and B. Bonet. *A Concise Introduction to Models and Methods for Automated Planning.* Morgan & Claypool Publishers, 2013.

[60] M. Ghallab, D. S. Nau, and P. Traverso. *Automated Planning and Acting.* Cambridge University Press, 2016.

[61] D. Hadfield-Menell, A. Dragan, P. Abbeel, and S. Russell. The off-switch game. *CoRR*, abs/1611.08219, 2016.

[62] D. Harel, G. Katz, R. Marelly, and A. Marron. First steps towards a wise development environment for behavioral models. *IJISMD*, 7(3):1–22, 2016.

[63] B. B. Hariri, D. Calvanese, M. Montali, G. De Giacomo, R. De Masellis, and P. Felli. Description logic knowledge and action bases. *J. Artif. Intell. Res. (JAIR)*, 46:651–686, 2013.

[64] Y. Hu and G. De Giacomo. Generalized planning: Synthesizing plans that work for multiple environments. In *IJCAI*, pages 918–923, 2011.

[65] Y. Hu and G. De Giacomo. A generic technique for synthesizing bounded finite-state controllers. In *ICAPS*, 2013.

[66] R. Hull, J. Su, and R. Vaculín. Data management perspectives on business process management. In *SIGMOD*, pages 943–948, 2013.

[67] IBM. An architectural blueprint for autonomic computing. IBM White Paper, 2005.

[68] E. Kharlamov, D. Bilidas, D. Hovland, E. Jimenez-Ruiz, D. Lanti, H. Lie, M. Rezk, M. Skjaeveland, A. Soylu, G. Xiao, D. Zheleznyakov, M. Giese, Y. Ioannidis, Y. Kotidis, M. Koubarakis, and A. Waaler. Ontology based data access in statoil. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2017. In print.

[69] B. Lacerda, D. Parker, and N. Hawes. Optimal policy generation for partially satisfiable co-safe LTL specifications. In *IJCAI*, pages 1587–1593, 2015.

[70] H. J. Levesque. On our best behaviour. *Artif. Intell.*, 212:27–35, 2014.

[71] H. J. Levesque. *Common Sense, the Turing Test, and the Quest for Real AI.* MIT Press, 2017.

[72] N. Lipovetzky and H. Geffner. Best-first width search: Exploration and exploitation in classical planning. In *AAAI*, pages 3590–3596, 2017.

[73] A. Marrella, M. Mecella, and S. Sardiña. Intelligent process adaptation in the smartpm system. *ACM TIST*, 8(2):25:1–25:43, 2017.

[74] A. Marron, B. Arnon, A. Elyasaf, M. Gordon, G. Katz, H. Lapid, R. Marelly, D. Sherman, S. Szekely, G. Weiss, and D. Harel. Six (im)possible things before breakfast: Building-blocks and design-principles for wise computing. In *ACM/IEEE D&P@MoDELS*, pages 94–100, 2016.

[75] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller. Playing atari with deep reinforcement learning. In *NIPS-2013 Workshop on Deep Learning*, 2013.

[76] P. G. Neumann. Trustworthiness and truthfulness are essential. *Communication of ACM*, 60(8):26–28, June 2017.

[77] F. Patrizi, N. Lipovetzky, G. De Giacomo, and H. Geffner. Computing infinite plans for LTL goals using a classical planner. In *IJCAI*, pages 2003–2008, 2011.

[78] A. Pnueli and R. Rosner. On the synthesis of a reactive module. In *POPL*, pages 179–190, 1989.

[79] A. Poggi, D. Lembo, D. Calvanese, G. De Giacomo, M. Lenzerini, and R. Rosati. Linking data to ontologies. *J. Data Semantics*, 10:133–173, 2008.

[80] F. Pommerening, M. Helmert, and B. Bonet. Higher-dimensional potential heuristics for optimal classical planning. In *AAAI*, pages 3636–3643, 2017.

[81] R. Reiter. *Knowledge in Action.* MIT Press, 2001.

[82] S. Russell, D. Dewey, and M. Tegmark. Research priorities for robust and beneficial artificial intelligence. *AI Magazine*, 36(4), 2015.

[83] S. Sardiña and G. De Giacomo. Realizing multiple autonomous agents through scheduling of shared devices. In *ICAPS*, pages 304–312, 2008.

[84] S. Sardiña and G. De Giacomo. Composition of congolog programs. In *IJCAI*, pages 904–910, 2009.

[85] S. Sardiña, G. De Giacomo, Y. Lespérance, and H. J. Levesque. On the semantics of deliberation in Indigolog - from theory to implementation. *Ann. Math. Artif. Intell.*, 41(2-4):259–299, 2004.

[86] S. Sardiña, G. De Giacomo, Y. Lespérance, and H. J. Levesque. On the limits of planning over belief states under strict uncertainty. In *KR*, pages 463–471, 2006.

[87] R. Seiger, S. Huber, and T. Schlegel. Toward an execution system for self-healing workflows in cyber-physical systems. *Software & Systems Modeling*, pages 1–22, 2016.

[88] J. F. Sequeda and D. P. Miranker. A pay-as-you-go methodology for ontology-based data access. *IEEE Internet Computing*, 21(2):92–96, 2017.

[89] Y. Shoham. Why knowledge representation matters. *Commun. ACM*, 59(1):47–49, 2016.

[90] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. P. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[91] S. Sohrabi, N. Prokoshyna, and S. McIlraith. Web service composition via the customization of golog programs with user preferences. In *Conceptual Modeling: Foundations and Applications - Essays in Honor of John Mylopoulos*, pages 319–334, 2009.

[92] M. Steinmetz and J. Hoffmann. State space search nogood learning: Online refinement of critical-path dead-end detectors in planning. *Artif. Intell.*, 245:1–37, 2017.

[93] J. Taylor, E. Yudkowsky, P. LaVictoire, and A. Critch. Alignment for advanced machine learning systems. Technical report, Machine Intelligence Research Institute, 2017.

[94] J. Torres and J. Baier. Polynomial-time reformulations of LTL temporally extended goals into final-state goals. In *IJCAI*, pages 1696–1703, 2015.

[95] M. Tsai, S. Fogarty, M. Vardi, and Y. Tsay. State of büchi complementation. *Logical Methods in Computer Science*, 10(4):1–27, 2014.

[96] W. van der Aalst, M. Pesic, and H. Schonenberg. Declarative workflows: Balancing between flexibility and support. *Computer Science - R&D*, 23(2):99–113, 2009.

b Curriculum Vitae

PERSONAL INFORMATION

Family Name, First Name:	De Giacomo, Giuseppe
Researcher ID:	orcid.org/0000-0001-9680-7658
Nationality:	Italian
Date of birth:	August 8, 1965
Home URL:	http://www.dis.uniroma1.it/~degiacomo

EDUCATION

1995	**PhD:** Dipartimento di Informatica e Sistemistica, Università degli Studi di Roma "La Sapienza", Italy Supervised by Prof. Maurizio Lenzerini.
1991	**Master:** Laurea in Ingegneria Elettronica (5 years), Facoltà di Ingegneria, Università degli Studi di Roma "La Sapienza", Italy
1990–1991	**Erasmus Visiting Student** at University of Bristol, UK, with John W. Lloyd, preparing master thesis

CURRENT POSITION

2006 – pres.	**Full Professor** (Professore Ordinario), Università degli Studi di Roma "La Sapienza", Italy

PREVIOUS POSITIONS

2001 – 2006	**Associate Professor** (Professore Associato), Università degli Studi di Roma "La Sapienza", Italy
1998 – 2001	**Assistant Professor** (Ricercatore), Università degli Studi di Roma "La Sapienza", Italy
1996 – 1997	**Research Associate**, University of Toronto, Toronto, ON, Canada, working with Prof. Hector Levesque and Prof. Ray Reiter
1993/1994	**Visiting Scholar**, Stanford University, Stanford, CA, USA, working with Prof. Yoav Shoham

FELLOSHIPS AND AWARDS

2016	**AAAI Fellow**, Fellow of the Association for the Advancement of Artificial Intelligence, Citation: *For significant contributions to the field of knowledge representation and reasoning, and applications to data integration, ontologies, planning, and process synthesis and verification.*
2015	**ACM Fellow**, Fellow of the Association for Computing Machinery, Citation: *For contributions to description logics, data management, and verification of data-driven processes*
2012	**ECCAI Fellow** (aka EurAI Fellow), Fellow of the European Association for Artificial Intelligence
2013	**Most influential paper in the decade** at th 11th International Conference on Service Oriented Computing (ICSOC'13) for the paper *Automatic Composition of E-services That Export Their Behavior* published at ICSOC03
2013	**Best paper award** at 7th international Conference on Web Reasoning and Rule Systems (RR'13) for the paper *Verification and Synthesis in Description Logic Based Dynamic Systems*
2013	**Miegunyah Distinguished Fellowship**, University of Melbourne, for the public lecture *Cognitive Robotics: The science of building intelligent autonomous robots and software agents*
2010	**IBM Open Collaborative Faculty Award 2010** for *Artifact-centric Business Process Modeling*, and **IBM Open Collaborative Faculty Award 2009** for *Radical Simplification of Artifact-Centric Buseness Process Modeling*

SUPERVISION OF YOUNG RESEARCHERS

He supervised a number of **PhD students**, including Daniela Berardi (PhD in 2005, now working in industry), Fabio Patrizi (PhD in 2009, now assistant professor at Sapienza), Riccardo De Masellis (PhD in 2013, now PostDoc at FBK, Trento, Italy) and Paolo Felli (PhD in 2013, PostDoc at U. Nottingham), as well as **PostDocs**. He mentored a number of **early career researchers**, including Sebastian Sardina (now associate professor at RMIT, Melbourne), Marco Montali (now associate professor at U. Bolzano, Italy), Stavros Vassos (now funder of the StartUp Helvia.io), and Sasha Rubin (Marie Curie Fellow at U. Napoli, Italy).

TEACHING ACTIVITIES

He has a wide teaching and academic experience. Since 1998, he has taught a large number of undergraduate and graduate courses at Sapienza. He has taught PhD courses in several European summer schools, including ESSLLI'03, ESSLLI'05, INFWEST'07. He has given several tutorials at top conferences, including ICSOC'04, WWW'05, ISWC'08, AAAI'10, IJCAI'15, IJCAI'16, ISWC'17.

ORGANIZATION OF SCIENTIFIC MEETINGS

He regularly serves as PC member of many international conferences and workshops in AI and CS, including IJCAI ('09, '07, '05, '03, '01, '99, '95), AAAI ('15, '10, '07, '06, '04, '02, '00, '98), KR ('16 '12, '10, '08, '06, '02, '00, '98, '96), ECAI ('16, '10, '08), ICAPS ('15, '11, '09, '08, '07, '06), AAMAS ('13, '12, '02), PODS ('17*, '15, '13, '11, '10, '09*, '08, '07, '03, '02, '00 - *PODS PC), ICDT ('15, '14, '11, '07, '05, '03, '99, '97), BPM ('12, '11, '10). Moreover he has been:

- Program Chair of KR'14
- Local organizer of KR'12
- Area chair of IJCAI'16, AAAI'12, IJCAI'11
- Senior PC of IJCAI'17, IJCAI'15, AAAI'16, AAAI'13, IJCAI'13, AAAI'11
- Workshops chair of AAAI'11, AAAI'10
- Organizer of 2003 Int. Workshop on Description Logics (DL'03) and of the 1st Cognitive Robotics Workshop at AAAI Fall Symp. (CogRob'98)

INSTITUTIONAL RESPONSIBILITIES

2016 – curr.	Member of the Department Directorate (Giunta di Dipartimento), Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza
2011 - 2014	Director of the PhD Program in Computer Science and Engineering, Sapienza
2003 - 2008	Scientific Coordinator of the Erasmus Program at the Engineering School, Sapienza

COMMISSIONS OF TRUST

- Artificial Intelligence J. (AIJ), Elsevier, Review Editor, since 2014; (Editorial Board Member, since 2013)
- Journal of Artificial Intelligence Research (JAIR), Associate Editor (2008 – 2015); (Editorial Board Member (2006 – 2008))
- Acta Informatica, Springer, Editorial Board Member, since 2015
- CoRR arXiv moderator for Artificial Intelligence, since 2014
- Vice President of Steering Committee of KR Inc., since 2016; in Steering Committee since 2014
- Member of W3C Recommendation Committee of OWL 2 Web Ontology Language Profiles (2012)
- Steering Committee of the Description Logics Workshop Series (2006 – 2009)
- AAAI Fellow Selection Committee, since 2017
- ECCAI/EurAI Fellow Selection Committee, since 2014

MEMBERSHIPS OF SCIENTIFIC SOCIETIES

Lifetime Member of Association for Computer Machinery (ACM)
Lifetime Member of Association for Advancement of Artificial Intelligence (AAAI)
Member of Italian Association for Artificial Intelligence (AI*IA)

Appendix: All on-going and submitted grants and funding of the PI (Funding ID)

On-going grants

Project	Funding source	Amount	Period	Role of the PI	Relation to current ERC proposal
ICE: Immersive Cognitive Environments	Sapienza	40 KEuro	2015-2018	PI	Application
Theory and Techniques for Reasoning about Actions and High-level Agent Control in Multi-agent Domains; NSFC Grant No. 61572535; Project PI: Yongmei Liu, Sun Yat-sen Univ. of Guangzhou	Natural Science Foundation of China	804,000 CNY	2016-2019	Sapienza PI	Algorithms
SHF: Small: Pushing the Frontier of Linear-Time Model-Checking Technology; NSF Grant No. 1319459; Project PI: Moshe Vardi, Rice University	NSF USA	304582 USD	2013-2018	Sapienza PI	Fundations

Grant applications

Project	Funding source	Amount	Period	Role of the PI	Relation to current ERC proposal
SafeWare: Platform for Hazard-centric IoT; H2020-IOT-2017:780761; Project PI: Lior Limonad, IBM Reasearch Haifa	EU H2020	4999370 Euro	2017–2020	Sapienza PI	Application
Beyond One-shot Planning: Cognitive Agent Planning Programs; ARC ID DP180102600;Project PI: Sebastian Sardina, RMIT, Australia	ARC Grant Australia	392541 AUD	2017-2020	Sapienza PI	Application

c Ten-year track record

PI's research concerns theoretical, methodological and applicative aspects in different areas of AI and CS, including: LTL and LDL over finite traces; Bounded situation calculus; Decidability of data-aware processes; Generalized planning by model checking and automata-theoretic techniques from Verification; ConGolog programming language based on Situation Calculus; Ontology Based Data Access (OBDA); DL-lite family: description logics with tractable data complexity; Reasoning on UML Class Diagrams; Service composition and synthesis; Regular path queries for view-based query answering in graph databases; Data integration with description logics constraints; Conjunctive query answering in description logics; Correspondence between description logics and logics of programs. This work has deeply impacted several areas of AI and CS.

IMPACT MEASURES

The PI is the author of more than 250 publications in top journals and conferences, including the following CORE A*/A conferences and journals (since 2007 in bold, career in italic): IJCAI (**17**, *22*, A*), AAAI (**11**, *16*, A*), KR (**10**, *20*, A*), AAMAS (**7**, *8*, A*), ICAPS (**5**, *5*, A*), PODS (**3**, *9*, A*), VLDB (**2**, *3*, A*), ISWC (**2**, *2*, A), CAiSE (**2**, *4*, A), ECAI (**3**, *4*, A), BPM (**3**, *3*, A), ICSOC (**1**, *4*, A), ICDT (**1**, *3*, A), Artif. Intell. (**4**, *7*, A*), J. Comput. Syst. Sci. (**2**, *3*, A*), Inf. Syst. (**1**, *2*, A*), J. Log. Comput. (**1**, *4*, A), J. Artif. Intell. Res. (**1**, *2*, A), J. Autom. Reasoning (**1**, *1*, A), ACM Trans. Comput. Log. (**1**, *2*, A), Theor. Comput. Sci. (**1**, *2*, A).

A comprehensive list of publication can be found on DBLP `http://dblp.uni-trier.de`. Currently the PI's 5 most cited papers are (citations from google scholar):

1. ***Tractable reasoning and efficient query answering in description logics: The DL-Lite family.*** *D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati. Journal of Automated Reasoning 39 (3), 385-429, 2007* - **1146** cit.
2. ***ConGolog, a concurrent programming language based on the situation calculus.*** *G. De Giacomo, Y. Lesperance, H. Levesque: Artif. Intell. 121(1-2): 109-169 (2000)* - **654** cit.
3. ***Linking data to ontologies.*** *A. Poggi, D. Lembo, D. Calvanese, G. De Giacomo, M. Lenzerini, R. Rosati. Journal on data semantics X, 133-173, 2008* - **560** cit.
4. ***Reasoning on UML class diagrams.*** *D. Berardi, D. Calvanese, G. De Giacomo. Artif. Intell. 168(1-2): 70-118 (2005)* - **547** cit.
5. ***Automatic Composition of E-services That Export Their Behavior.*** *D. Berardi, D. Calvanese, G. De Giacomo, M. Lenzerini, M. Mecella. ICSOC 2003: 43-58* - **519** cit. Awarded as "**the most influential ICSOC paper in the last 10 years**" at ICSOC 2013.

According to Google Scholar, July 2017, the PI's h-index is 68, with 16952 citations and his i10-index is 191. These values are among the highest in Europe in AI and CS and make the PI the 3rd most cited CS author in Italy, according to a study available at `http://via-academy.org`.

REPRESENTATIVE PUBLICATIONS (SELECTION)

1. ***Generalized Planning: Non-Deterministic Abstractions and Trajectory Constraints.*** *B. Bonet, G. De Giacomo, H. Geffner, S. Rubin. IJCAI 2017* - CORE A*.
2. ***Agent planning programs.*** *G. De Giacomo, A. Gerevini, F. Patrizi, A. Saetti, S. Sardina. Artif. Intell. 231: 64-106 (2016)* - CORE A*.
3. ***Bounded situation calculus action theories.*** *G. De Giacomo, Y. Lesperance, F. Patrizi. Artif. Intell. 237: 172-203 (2016)* - CORE A*.
4. ***Regular Open APIs.*** *D. Calvanese, G. De Giacomo, M. Lenzerini, M. Vardi. KR 2016: 329-338.* - CORE A*.
5. ***Synthesis for LTL and LDL on Finite Traces.*** *G. De Giacomo, M. Vardi: IJCAI 2015: 1558-1564* - CORE A*.
6. ***Data complexity of query answering in description logics.*** *. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati. Artif. Intell. 195: 335-360 (2013)* - CORE A*.
7. ***Automatic behavior composition synthesis.*** *G. De Giacomo, F. Patrizi, S. Sardina. Artif. Intell. 196: 106-142 (2013)* - CORE A*.
8. ***Verification of relational data-centric dynamic systems with external services.*** *B. Bagheri Hariri, D. Calvanese, G. De Giacomo, A. Deutsch, M. Montali. PODS 2013: 163-174* - CORE A*.

9. ***On simplification of schema mappings.*** *D. Calvanese, G. De Giacomo, M. Lenzerini, M. Vardi. J. Comput. Syst. Sci. 79(6): 816-834 (2013)* - CORE A*.
10. ***View-based query answering in Description Logics: Semantics and complexity.*** *D. Calvanese, G. De Giacomo, M. Lenzerini, R. Rosati. J. Comput. Syst. Sci. 78(1): 26-46 (2012)* - CORE A*.

INVITED TALKS (SELECTION)

– On **LTL and LDL on Finite Traces: Reasoning, Verification, and Synthesis**: GenPlan@ICAPS 2017 (Keynote), Pittsburgh, USA; SR 2016 (Keynote), New York, USA; Highlights of Logic, Games and Automata 2015 (Keynote), Prague, CZ; ICAPS 2013 (Keynote), Rome, IT.
– On **Data-Aware Business Processes**: York U., (Distinguished Lassonde Lecture), Canada, 2017 ; U. of Toronto, Canada, 2017; WS-FM:FASOCC@BPM 2014, Eindhoven NL; ECAI 2014 (Keynote for Frontiers of AI), Prague, CZ.
– On **Bounded Situation Calculus**: WS. Formal Methods in AI 2017 (Keynote), U. "Federico II", Naples, IT; TIME 2015 (Keynote), Kassel, GE; WS. HYBRIS 2015 (Keynote), Potsdam, GE.
– On **Service Composition and Synthesis**: ICSOC 2013 (Invited talk for prize as most influential paper of decade), Berlin, GE; U. Brescia, IT 2012; U. of Toronto Canada, 2010; York University, Toronto, Canada, 2010; INFINT WS 2009, Bertinoro, IT; MSI 2005 Berlin, GE , Caen, FR.
– On **Ontology-based Data Access and Integration**: DL 2013 (Keynote), Ulm, Germany; U. of Toronto, Canada, 2010; Semantic Days Conference 2009, Stavanger, NO; IBM Research Center Watson, Hawthorne, NY, USA 2008.

RESEARCH PROJECTS (SELECTION)

The PI has leaded several projects, including: (2005–2008) EU FP6-7603 **TONES: Thinking ONtologiES**, PI for Sapienza, value: EUR 264,000 (total value: EUR 1,438,910), from final review: *The TONES project can be considered as a success story of a FET-project in terms of scientific achievements*; (2010–2013) EU FP7-ICT-257593 **ACSI: Artifact-Centric Service Interoperation**, PI for Sapienza, and Scientific Coordinator for the wole project, value: EUR 435,000 (total value: EUR 3,243,937), from final review: *The scientific productivity of ACSI has been extraordinary, leading to a great impact on BPM, DB and AI research, as indicated by invited talks and tutorials on ACSI results in many high profile conferences, and by the addition of ACSI topics to the list of conference and workshop topics in these areas.*; (2012–2016) EU FP7-IST-IP-318338 **OPTIQUE: Scalable End-user Access to Big Data**, key personel of Sapienza, value: EUR 802 488 (total value: EUR 9,838,329); (2009–2014) Open Collaboration Research W0954341 with Rick Hull of IBM T. J. Watson Research Center, NY, on **Data Aware Business Processes and Operation, through An Artifact-Centric Approach**, PI, value: USD 45,000; (2010–2012) UK Royal Society International Joint Project 2009/R2 on **Web Services Automatic Synthesis through ATL Symbolic Model Checking**, with Alessio Lomuscio, Imperial College London, total value: GBP 12,000; (2012–2014) Australian Research Council (ARC) Discovery Project DP120100332 **Optimisation of Embedded Virtual Complex Systems by Re-using a Library of available component**, with Sebastian Sardina of RMIT and Maurice Pagnucco of Univ. of Sidney, PI for Sapienza, total value: AUD 406,278; (2013–2015) Sapienza Ateneo Project **Spiritles: Spiritlet-based Smart Spaces**, PI, value: EUR 60,000. Currently active project are listed in the CV.

INDUSTRIAL LEADERSHIP

A pioneer project in 2003 with IBM Tivoli Lab highlighted several fundamental limitations in using semantic technologies available at that time for knowledge management. This gave rise to a novel research line that led to a new kind of Description Logic, called DL-Lite, and a new paradigm for accessing data through semantic technologies, called Ontology-based Data Access (OBDA). Since then, the PI led several industrial explorations promoting OBDA for data integration, preparation and discovery, which involved private and public organizations, such as the Monte dei Paschi di Siena Bank, Italian Minstery of Finance and Economics, Telecom Italia, Bloomberg, Italian Automobil Club (current). The success of such explorations has led the PI to found the Sapienza Start Up **OBDA Systems** (`http://www.obdasystems.com`) in Feb. 2017. The PI has been also a Contributor to W3C recommendation of OWL 2 Web Ontology Language Profiles (`https://www.w3.org/TR/owl2-profiles/`).