Game-theory strategies for open-source Infrastructure-as-Code

Alfonso E. de la Fuente Ruiz
European Science Foundation
Valencia, Spain
alfonsodelafuenteruiz@yahoo.es

Galia Novakova Nedeltcheva
Politecnico di Milano, DEIB
Milan, Italy
galia.nedeltcheva@polimi.it

***Abstract*— Infrastructure-as-Code can be designed, developed, and managed with an Open-Source Software approach. While openly embraced by academia, the OSS philosophy typically elicits some skepticism among industrial partners who want to ensure that adopting it will not enable their market competitors or harm them in any way, like by releasing valuable intellectual property into the wild.**

In this paper, we study this phenomenon from the perspective of game-theory, to then draw conclusions on what information to share, when, why, how, and with whom, so that risks are minimized, and benefits are maximized for each, and every partner engaged in IaC-focused collaborative projects that make use of an OSS approach.

***Keywords*— Infrastructure-as-Code (IaC), Open-Source Systems, Game theory, Business strategy**

I. Introduction

Infrastructure-as-Code (IaC) has become one of the most prominent paradigms in the Cloud Computing (CC) business. Like any piece of software, IaC can be designed, developed, and managed with an Open-Source Software (OSS) approach. When industrial partners are required to collaborate among themselves, or with academic, and other publicly-funded partners, some resistance becomes often apparent among them on what to share with others and why, which then turns into a bottleneck from a knowledge management perspective [1], therefore degrading collaboration, and impairing the results, performance, quality, and success perspectives of cooperative projects. In game-theory, there are two main classes of games: non-cooperative (the default scenario) and cooperative (where players are allowed to cooperate for mutual benefit), which include the study of positive-sum games or so-called 'win-win' scenarios. We will show under which specific circumstances can players be benefited or harmed by sharing IaC-related information with one or more other players.

This paper studies the potential of the game-theorical approach for collaborative Open Source systems (OSS) development for IaC. As such, we investigate the potential of collaboration between market players in the context of cloud computing systems, particularly regarding the release of IaC blueprints, routines, datasets, and schematics openly (online or in an intranet) under an open license and/or as OSS. For this purpose, we will identify the background context in information technology markets nowadays, and finalize with a game-theoretical approach to collaboration based on information-sharing that is grounded on the DevOps paradigm and illustrated with some examples.

One paradigmatic example of OSS usage is the field of cybersecurity, where any actor can contribute a patch for a recently discovered flaw or exploit (0-day vulnerability). This enhances security because there are more eyes looking into the source code and its flaws, who are then in turn able to propose solutions in a distributed and collaborative manner, from various local, specialized perspectives, and fields of expertise. This level of distributed collective intelligence [2], [3], [4] would be nearly unattainable for closed-source applications, even if development teams had thousands of expert agents involved.

Cybersecurity, interoperability of hardware, and international collaboration, are some of the most prominent fields of application of Open Data (OD) and OSS. One recent example of collaboration is in the medical research domain, where papers are shared openly among researchers for enhanced results in healthcare, like for instance they were, during the latest coronavirus pandemic period, as evidenced by a statement issued on the 31st of January 2020 from publishers of subscription journals.

II. Background

A. Infrastructure-as-Code

IaC enables the automation of provisioning, deployment, configuration and management of computational infrastructure (such as servers and virtual machines) through source code instead of classical manual processes. In the cloud computing environment, IaC has a lot of potential as a significant time-saver when an application needs to be redeployed on a different set of resources, or running on different infrastructural tools [5], [6], [7], [8] or even onto more decentralized models such as the Cloud Edge. It also saves time when working with Digital Twins, which replicate an asset in a virtual environment, and with the spiral model for software development with incremental prototyping [9], [10], which has inspired the adaptation of modern methodologies like Agile [11].

B. Open-Source Systems

In recent years, Open-Source (OS) has become pervasive throughout the ICT sector. OS needs to be considered as an alternative nowadays when designing the business strategy of any ICT company [12]. On the other hand, OS hardware is, in terms of adoption, in a similar path as OSS.

The adoption of cloud computing operational models by ICT industrial players can accelerate the commoditization of OSS. According to a study carried out by the European Commission [13], several OSS solutions have been developed to solve specific challenges that can be found in the cloud continuum service stack. Cloud vendors, especially the larger ones, are monetizing OSS by integrating it into their own cloud services-based proprietary derivatives, but few among them release these as OS.

FIST2023: 2nd International Workshop on the Foundations of Infrastructure Specification and Testing
L'Aquila, Italy, March 13-17, 2023 (Submitted in December 2022)

C. Open-Data and Open-Data Applications

Open-data (OD) is one of the main objectives of e-government policies, funding agencies, and researchers [14]. Various value-centric business model frameworks for managing ODA have been devised [15]. ODA are naturally very dependent on the availability and integration of OD with their routines, which, in turn add the most value when they are based on OSS as well.

We will first focus on open-sourcing strategies for IaC development for Cloud systems, illustrating it with examples. There are two main areas of concern in this context: OSS applications, and Open Data Applications (ODA).

D. Game theory

Game theory is the study of mathematical models of strategic interactions among rational agents (or "persons"), originally addressing two-person zero-sum games, and later extended to include more general game models [16], [17].

Markov Decision Processes (MDP) are stochastic control processes that evolve in discrete time intervals and provide a framework for modeling decision-making under uncertainty (particularly for decision-making in cloud computing [18]).

Stochastic games generalize these to multiple interacting decision makers, as well as to strategic-form simultaneous games for dynamic situations in which the environment changes in response to the players' choices. When a game is repeated numerous times, it is called an iterated, or multi-stage game. Unlike a game played once, a multi-stage game allows for any strategy to be contingent on past moves, thus taking into account reputational effects, bargaining power, and retribution. The number of iterations can be increased arbitrarily to account for the smallest measurable time intervals.

III. THE BUSINESS CONTEXT OF OPEN-SOURCING INFRASTRUCTURE-AS-CODE

The maximum value from collaborative activities is generally thought to be obtained when mixing Open-Data (OD) with open algorithms into ODAs. This is because not only any actor can retrieve (and sometimes contribute) datasets, but also any actor can contribute to the processing of these datasets in the best way, by studying the data structures and algorithms, and then proposing improvements to ODA's routines. When an application releases the source code of their algorithms publicly under some form of open license, we consider that application to be an example of OSS. Open routines are especially relevant in Machine Learning Operations (MLOps), because these make use of formulas and algorithms whose patenting is forbidden by law, and because societies and governmental bodies demand features such as unbiasedness, and explainability (e.g.: in AI) [11].

Furthermore, the users of ODA can rest assured that these applications are functioning in the way they are supposed to (e.g.: without any harmful backdoors), since everyone can read the source code if they so choose. This openness in both data and source code is oftentimes a legal prerequisite for the e-government application domain, and others, as for instance, in the context of systems reliability, distributed systems [19] [20], supply chain management, technological sovereignty and technological neutrality with regards to vendor choice.

A concrete example of OSS in cybersecurity is OpenPGP, a non-proprietary format for authenticating or encrypting data that uses public key cryptography, which claims to be the most widely used e-mail encryption standard. OpenPGP formats and uses are specified in several Request-for-comment (RFC) documents by the Internet Engineering Task Force (IETF), which makes voluntary standards that are often adopted by Internet users, network operators, and equipment vendors, thus helping shape and direct the development of global networks. A specific example of OSS for IaC is HashiCorp's Vault which is OS and cloud-agnostic and where Secrets Engines are executed in the Cloud. There is no argument about the advantages that open standards provide for the development of both hardware and software, like for instance in the design and commercialization of communication ports in computers and smartphones, such as RJ-45 or USB.

There are various different kinds of collaborative creation that may involve attribution and/or payment of royalties, for instance, as described in the various types of Creative Commons licenses that were inspired by the GNU General Public License (GNU-GPL), which is at the basis of most OSS licenses and modern Linux-based systems. In turn, various large organizations have released their own versions of open licenses, for instance NASA, with their Open Source Agreement [21], CERN, with their Open Hardware License [22], or University of California in Berkeley's RISC-V [23], an open, royalty-free standard for Instruction Set Architectures (widely adopted, e.g.: in IoT microelectronics). All of these do foster international collaboration, enhance productivity gains, and facilitate the interoperability of systems, particularly of software systems such as firmware or inter-connected Cloud Systems [24]. Nonetheless, it is certainly the computer science and informatics fields where these open standards are most widely known and adopted these days.

All these paradigms for collaboration are in fact so successful that regulators worldwide were forced to develop Competition Law by imposing antitrust measures [25] and regular conflict-of-interest checks in order to deter various ways of collusion between prominent market players, be this collusion tacit or not, as in price coordination practices by oligopolies, concerted action, and conscious parallelism. Those naturally result in undesirable side-effects such as cartelization, and monopolistic practices.

However, the industrial sector, and some governments, have always been wary (sometimes quite literally) about sharing knowledge and data, based on the belief that their competitors or adversaries may gain unfair advantage from doing so. This has resulted in convoluted intellectual protection practices, and sophisticated information gathering systems, sometimes resulting in the escalation of conflicts, e.g., espionage at the national and industrial levels, or also the so-called "patent-wars" [26] [27] [28].

Hence, we shall now study, from a game-theoretical perspective, whether there are any advisable substitute strategies that may be superior to those other, more pernicious ones, in terms of value creation, both for individual players, and for the market as a whole [29].

IV. OPEN-SOURCE STRATEGIES FROM A GAME THEORY PERSPECTIVE

In game theory [16] [30] as applied to markets, the simplest kind of game G is given by a set of two players,

$\{u_0, u_1\}$ among which one of them (u_0) is the environment, understood in the widest sense, which reacts to the actions of the 'real' player and conditions their next moves. Typically, more than one 'real' player is introduced ($u_1, \dots, u_n$) in order to study market dynamics, and for each of these players, we introduce a tuple of values s_i which corresponds to the set of actions that one specific player executes during a play. We call this tuple a strategic profile or *strategy profile*.

In a given game $G = \{S_1, \dots, S_n; u_1, \dots, u_n\}$, where $(S_1, \dots, S_n)$ is the space of all possible strategies, we say that the strategy profile $s = (s_1, \dots, s_i, \dots, s_n)$ is *weakly dominated in Pareto sense* by another strategy profile $s' = (s'_1, \dots, s'_i, \dots, s'_n)$ if, and only if the inequality $u_i(s'_1, \dots, s'_n) \geq u_i(s_1, \dots, s_n)$ is fulfilled, for each player u_i, for every combination of pure strategies, and for any one (but not all) of them in strict sense, according to their *payoffs* during the next turn [16].

On the contrary, a strategy profile is *strictly dominated* when the previous inequality is fulfilled in strict sense for every combination of strategies of the other players. The payoffs are in turn defined by a von-Neumann-Morgenstern utility function, which considers that any one player may be risk-averse, meaning that an individual might refuse a fair gamble with an expected value of zero.

Finally, a strategy profile $s = (s_1, \dots, s_i, \dots, s_n)$ is defined as a *Pareto optimum* (or it would be said to be "*Pareto efficient*") if, and only if, it is not *dominated* by any other strategy profile. Naturally, every player will want to attain Pareto optimality for their strategy profile, since that one results in better payoffs and thus yields the most value (as measured e.g.: in terms of economic value, revenue, or productivity).

For evaluating the efficiency of a strategy profile in a repeated multiplayer game, where at least one player u_1 releases information to all other players, and at least one player u_2 does not release any information to any other players, we need to evaluate the strategy's dominance in Pareto sense.

We shall then define the following two strategic profiles:

- $s_o = (s_{o_1}, \dots, s_{o_i}, \dots, s_{o_n})$ is the "open" strategy of a player releasing information to all other players, which they may then use for their own advantage, or for the advantage of other players, during their subsequent plays.
- $s_c = (s_{c_1}, \dots, s_{c_i}, \dots, s_{c_n})$ is the "closed" strategy of a player not releasing any information to any other player, ever.

And then evaluate, for every combination of strategies of the other players, when the other inequalities are fulfilled:

- $\forall i: u_i(s_{o_1}, \dots, s_{o_n}) \leq u_i(s_{c_1}, \dots, s_{c_n})$, meaning that the payoff of the "closed" strategy profile is at least as good as that of the "open" strategy for any one player.
- $\forall i: u_i(s_{o_1}, \dots, s_{o_n}) \geq u_i(s_{c_1}, \dots, s_{c_n})$, meaning that the payoff of the "open" strategy profile is at least as good as that of the "closed" strategy for any one player.

For the sake of simplicity, we assume that every player u_i is equivalent and interchangeable with each other. This might, however, not always be the case; for instance, it might be less risky in average for a large corporation to release information to other players than it would be for a small business (since these can do less harm as measured in market share or percentage of revenue, for instance), or in reverse (since small players tend to be more agile than larger ones and thus able to react more rapidly to disclosures). We also leave aside the cases in which any one player releases information only to a finite set of other players, because those special cases can be derived from the general case.

One typical example of the general case would be releasing source code online about IaC deployments in an online platform such as GitHub (where, in principle, all other players can access the information, given that they are aware of its existence, location, and utility). An example of the special case would be the use of a collaboration platform sitting in an intranet (e.g.: SharePoint [31], where a set of players share information privately, e.g. about Ansible, an open source community project for automation and orchestration of cloud infrastructure), so that in theory it does not escape the platform in any manner that may be advantageous to other players.

Now it is straightforward to evaluate when, for every combination of strategies of other players, the following equality is fulfilled:

- $\forall i: u_i(s_{o_1}, \dots, s_{o_n}) = u_i(s_{c_1}, \dots, s_{c_n})$ (Eq. 1)

That happens only when the value for other players of the information that is being released is equal to zero, meaning that no other player will use the information being released as a competitive advantage during their next moves, but also that the information being released does not hamper their payoffs during their successive moves.

One way in which players can share relevant data in ODAs while preventing valuable insider information from getting out is to make use of synthetic datasets (data created artificially but in the same format and patterns as real data) or anonymized datasets (data which has been sanitized so to make it impossible to identify what or whom it refers to).

The case of intentional deceit in collaborative OSS development has been both widely and deeply studied, e.g.: in the context of the well-known thought experiment of the '*prisoner's dilemma'* [17], and for it, various strategies have been devised, like "*tit-for-tat*", either with, or without a *forgiveness threshold*. Deriving an optimal strategy in game theory can be done by either computing Bayesian Nash equilibria, or by adopting the "frequentist approach" of launching numerous experimental surveys (e.g.: with "*Monte Carlo*" simulations [32]) and then computing optimal strategies statistically. Most software firms who approach this challenge in a non-rigorous manner are likely doing the latter, basing their decision-making processes on sheer personal experience alone [33] (this generally being less effective than a rigorous approach).

When the processing capacity of any one player becomes saturated, they have two choices: they can either completely ignore the information being released by one, some, or all players, or they may try to process it during their next rounds in the game, therefore losing aggregated value, because their strategy profiles must defer other productive actions in order to account for additional information-processing steps.

Now let us explore the remaining two cases:

- $\forall i: u_i(s_{o_1}, \dots, s_{o_n}) < u_i(s_{c_1}, \dots, s_{c_n})$ (Eq. 2)

This means that the value of the "open strategy" is smaller than the value of the "closed strategy". It is evident that, if all players use a fully closed strategy, there is no room left at all for collaboration (and not even for communication). Therefore, this inequality is never true then; at most, their payoffs must be equal, even during direct confrontation or all-out war, unless at least one player is releasing deceitful information and at least one other player is candidly trusting it (or "buying it"). Naturally, for collaboration to occur, the conditions for an open and frank dialogue must be established first, and incentives must appear with clarity to all sides of the partnership.

The trickiest case is when one or more players are releasing information of value, and one or more other players are making use of it for their own advantage. In those cases, we must evaluate where the larger gains are, in average, and over the whole duration of the game, by adding all aggregated value and then subtracting all harms (lost value), that is, calculating the overall balance of payoffs. In game theory, this "overall balance" is computed from the *Shapley value* as applied to a coalition of players. The Shapley value function (which helped earn the 2012 Nobel Prize in Economic Sciences) also considers the bargaining power [34] of individuals (e.g.: threatening to destroy value for others unless some requisitions are met by other players, effectively blackmailing them, or, some would say, subduing or 'incentivizing' them), which is in turn constrained by the dominance of strategies (unless in cases of 'bluff').

The last case we need to evaluate is:

- $\forall i: u_i(s_{o_1}, \ldots, s_{o_n}) > u_i(s_{c_1}, \ldots, s_{c_n})$ (Eq. 3)

This means that the "closed strategy" is *strictly dominated* by the "open strategy" for every combination of strategies of the other players, after computing the balance between the harm being suffered and the benefits being reaped, in Shapley's sense [11]. That is what we think it to be the case for collaborative OSS development, under our set of assumptions and with the precautions stated above, and it means that open collaborative strategies have more value than closed ones.

V. Conclusions

Leaving Shapley aside by assuming that, in the long term, the average payoffs of having collaboration are higher than those of not having any (which is a self-evident argument for any intelligent species), the implications of the last inequality as stated above are numerous from the business perspective. For instance, if any player wants to obtain the most value from collaborative endeavors like OSS development of IaC:

- The player using an open strategy should release all relevant information that is useful for collaboration with any other collaborating player, whenever their collaboration framework is safeguarded, even when this information is not of any value to the party who releases it. Typically, the less relevant information they share, the smaller the aggregated gains are.
- The player should not release any information which may saturate their collaborators, that is, all the information released is 'pertinent' and not obsolete.
- The player should not release any information that may be deceitful (either accidentally or intentionally) unless in an all-out war scenario, which effectively makes cooperation vanish and results in a classical non-cooperative game, where maximum market value is impossible to be reached in most real-world scenarios.
- The player should not release any information that any other players who are outside of their collaboration framework may use for their own advantage, unless these other players are guaranteed to be altruistic, and to maintain confidentiality with respect to all other non-altruistic and/or non-confidential players.
- Players in the context of fierce competition, and under the possibility of deceit, should calculate the overall balance that collaboration brings to each of them, and to all other parties, in order to decide their strategy. These calculations on all other parties imply accounting for all u_0-externalities in the economic sense [35]; e.g: environmental pollution, brand damage, or social unrest. They may do that by computing the optimal Bayesian equilibrium (in Nash's sense), and/or based on market surveys and previous experience, ideally including forecasts of expected returns by involving Shapley values into their calculations.
- Players should avoid pernicious practices that do not foster win-win collaboration, but rather, seek to gain advantage at the expense of other players, by preventing getting involved into negative-sum games where the only way for one party to prosper is at the expense of others.

Unsurprisingly, most of these conclusions are fully consistent with commonly accepted business ethics, although, inevitably, some real-world markets tend to drift towards non-cooperative scenarios. As markets evolve, they may eventually reach an equilibrium state where every player does collaborate to the best of their abilities (maximum yield), or, on the contrary, they may orbit back and forth through a 'trough of disillusionment' where some, or all players are competing more fiercely. This latter case often results in monopolies and mergers [25] that drive their competitors out of the market, since when two players merge, from a game theoretical perspective, they start to share (ideally) all information between their inner components (departments or factions). Therefore, these large conglomerates become more efficient than smaller players, due not only to their larger size but to better knowledge management practices, which also imply more effective and actionable strategic profiles, on average, as well as higher-quality service provision, and better market leverage. From a formal point of view, the only difference between the payoffs of an infinitely collaborative market with multiple players, or those of a market with only one giant monopoly, is their capacity to respond to the challenges of the environment, u_0. Experience tells us that, from an evolutionary perspective, adaptive diversification is generally less risky and better suited for long-term survival in an uncertain environment.

VI. Acknowledgments

This work was partially funded by the EU Commission in the Horizon 2020 research and innovation program under grant agreement No. 101000162.

VII. References

[1] Miltiadis D. Lytras, Ronald Maier, Ambjörn Naeve, et al., "The Current Bottleneck of Knowledge Management and How Information Technology can be Successfully Used to Reduce it," in *Knowledge Management Strategies*, IGI Global, 2008, pp. 289-315.

[2] Eneko Osaba, et al. "Community detection in networks using bio-inspired optimization: Latest developments, new results and

perspectives with a selection of recent meta-heuristics," *Applied Soft Computing,* vol. 87, no. 2, 2020.

[3] Swarm Intelligence in Data Mining. Ajith Abraham, Crina Grosan, Vitorino Ramos (Eds.). s.l. , "Studies in Computational Intelligence," *Springer,* vol. 34, 2006.

[4] Blum, Christian, and Daniel Merkle, eds. , "Swarm intelligence: introduction and applications," *Springer Science & Business Media,* 2008.

[5] Alonso, Juncal, et al., "Piacere: Programming trustworthy infrastructure as code in a secure framework," in *CEUR Workshop Proceedings. CEUR-WS*, 2021.

[6] Artac, M., Borovssak, T., Di Nitto, E., Guerriero, M., & Tamburri, D. A., "DevOps: introducing infrastructure-as-code," in *IEEE/ACM 39th International Conf. on Software Engineering Companion (ICSE-C),* 2017.

[7] Artac, M., Borovšak, T., Di Nitto, E., Guerriero, M., Perez-Palacin, D., & Tamburri, D. A., "Infrastructure-as-code for data-intensive architectures: A model-driven development approach," in *Proceedings of IEEE International Conference on Software Architecture (ICSA)*, 2018.

[8] Indika Kumara, Martín Garriga, Angel Urbano Romeu, Dario Di Nucci, Fabio Palomba, Damian Andrew Tamburri, Willem-Jan van den Heuvel, "The do's and don'ts of infrastructure code: A systematic gray literature review," *Information and Software Techn.*, vol. 137, 2021.

[9] Roger S. Pressman, Bruce R. Maxim, "Software Engineering: A Practitioner's Approach", (1st ed.). McGraw-Hill College, 1991.

[10] Boehm, Barry, et al., "Using the WinWin spiral model: a case study," *IEEE Computer,* vol. 31, no. 7, pp. 33-44, 1998.

[11] E. Raj, "Engineering MLOps", Packt Publishing, 2021.

[12] R. A. Ghosh, "Study on the:Economic impact of open source software on innovation and the competitiveness of the Information and Communication Technologies (ICT) sector in the EU," MERIT, 2006.

[13] EC, "“The impact of Open Source Software and Hardware on technological independence, competitiveness and innovation in the EU economy”. Final Study Report," Brussels, 2021.

[14] Georg J.P. Link, Kieran Conboy, Debora Jeske, Matt Germonprez, Sean P. Goggins, Joseph Feller, et al., "Contemporary Issues of Open Data in Information Systems Research: Considerations and Recommendations," *Communications of the Association for Information Systems,* vol. 41, no. 1, 2017.

[15] C.-C. Yu, "A value-centric business model framework for managing," *Journal of Organizational Computing and Electronic Commerce,* no. 26, pp. 1-2, 80-115, 2016.

[16] Joaquín Pérez, José Luis Jimeno, Emilio Cerdá, "Teoría de Juegos", Pearson-Prentice Hall, 2004.

[17] Julie Kendall, Kenneth E. Kendall, Matt Germonprez, "Game theory and open source contribution: Rationale behind corporate participation in open source software development," *Journal of Organizational Computing and Electronic Commerce*, vol. 26, no. 4, pp. 323-343, 2016.

[18] H. M. Dipu Kabir, Abbas Khosravi, Subrota K. Mondal, Mustaneer Rahman, Saeid Nahavandi, Rajkumar Buyya, "Uncertainty-aware Decisions in Cloud Computing: Foundations and Future Directions," *ACM Computing Surveys*, vol. 54, no. Article 74, 2021.

[19] Daniel Sun, Shiping Chen, Guoqiang Li, Yuanyuan Zhang, Muhammad Atif, "Multi-objective Optimisation of Online Distributed Software Update for DevOps in Clouds," *ACM Transactions of Internet Technology,* vol. 19, no. 3, Article 43, 2019.

[20] Krcmar, Felix Willnecker Helmut, "Multi-Objective Optimization of Deployment Topologies for Distributed Applications," *ACM Trans. Internet Technol.,* vol. 18, no. 2, Article 21, 2018.

[21] "NASA Open Source Development and Agreements," [Online]. Available: https://www.nasa.gov/open/open-source-development.html.

[22] "CERN Open Hardware Licence," [Online]. Available: https://ohwr.org/cernohl.

[23] "Reduced Instruction Set Computer V," [Online]. Available: https://riscv.org/.

[24] Kiranbir Kaur, Dr. Sandeep Sharma, Dr. Karanjeet Singh Kahlon, "Interoperability and Portability Approaches in Inter-Connected Clouds: A Review," *ACM Comput. Surveys,* vol. 50, no. 4, Article 49, 2017.

[25] M. E. Porter, "Competition and antitrust: Toward a productivity-based approach to evaluating mergers and joint ventures," *The antitrust Bulletin,* vol. 46, no. 4, 2001.

[26] J. Wise, "Patent wars: Affordable medicines versus intellectual property rights," *BMJ Clinical Research,* no. 348, 2014.

[27] F. Z. Yongwook Paik, "The Impact of Patent Wars on Firm Strategy: Evidence from the Global Smartphone Industry," *Organization Science* , vol. 27, no. 6, 2016.

[28] Laurent R Bergé, Thorsten Doherr, Katrin Hussinger, "How patent rights affect university science," *Industrial and Corporate Change,* 2022.

[29] I. Murphy, "Everyone Wins With Open Source Software," 13 September 2016. [Online]. Available: https://www.linux.com/news/everyone-wins-open-source-software/.

[30] Rubinstein, Martin J. Orborne and Ariel, "A course in game theory", MIT Press, 1994.

[31] Oscar Medina, Ethan Schumann, "DevOps for SharePoint with Packer, Terraform, Ansible, and Vagrant", Apress Media LLC: Welmoed Spahr, 2018.

[32] J. Geweke, "Bayesian inference in econometric models using Monte Carlo integration," *Econometrica, by the Econometric Society,* vol. 57, no. 6, pp. 1317-1339, 1989.

[33] D. H. Steph Subanidja, "The influence of the knowledge management “bottleneck” on a company's performance," *Management & Marketing. Challenges for the Knowledge Society* , vol. 12, no. 3, 2017.

[34] M. E. Porter, "How competitive forces shape strategy," *Harvard Business Review,* pp. 137-145, 1979.

[35] C. I. Jones, "The Costs of Economic Growth," Stanford University, 2008.