Conceptual Modeling to Support the "Larger Goal" Pivot – An Example from Netflix

Vik Pant[1] and Eric Yu[1, 2]

[1]Faculty of Information, University of Toronto, Toronto, Canada
[2]Department of Computer Science, University of Toronto, Toronto, Canada
vik.pant@mail.utoronto.ca
eric.yu@utoronto.ca

Abstract. Many organizations mistakenly or inadvertently focus on tactical aims rather than on strategic goals. "Strategy" commonly denotes long-term objectives and high-level policies while "tactic" refers to deployment concerns and implementation considerations. By focusing on lower-level objectives an organization can potentially overlook or neglect better ways of achieving higher-level goals. Shifting from a short-term to a long-run orientation can be considered a type of pivoting, as the structure and relationships of an organization are substantially reconfigured. The Larger Goal pivot is essential when lower-level options for achieving a higher-level organizational goal are either unavailable or insufficient. It entails shifting focus to a larger or higher goal and exploring strategic alternatives to satisfy that goal. In this paper we present conceptual models of the Larger Goal pivot based on a historic example from Netflix – a movie streaming service.

Keywords: Pivoting. Design. Analysis. Modeling. Strategy. Tactic.

1 Introduction

The distinction between strategy and tactic is studied by researchers in many disciplines including economics and business management [1]. The term "strategy" denotes long term objectives and high level policies while the term "tactic" refers to deployment concerns and implementation considerations [2]. It is argued that ideally tactics should support the achievement of their associated strategies [3]. However, in the business world, this is not always observed to be the case. Many organizations, startups and large enterprises alike, mistakenly or inadvertently center their plans and actions around tactics rather than around strategy. This is problematical for them because even if they can meet their short-term targets – the fulfilment of their long-term goals is far from guaranteed.

Organizations can pivot and shift focus from a short-term to a long-run orientation. For example, Microsoft pivoted away from defending the market share of Windows operating system (OS) from threats by rival Linux to building application software that could run on multiple operating systems [4, 5]. This pivot allowed Microsoft to access the Linux installed base and increase the addressable market for its

applications at the cost of losing some OS market share. eBay pivoted away from being an online auctioneer to becoming a diversified eCommerce platform on the Internet [6, 7]. This pivot positioned eBay to compete in many new markets including those served by Amazon while moving away from rivals in its original market. In spite of many success stories associated with pivoting – it is a nontrivial undertaking that requires foresight and insight about the nature and scope of the intended change.

The notion of pivoting was popularized among entrepreneurs, startup founders, and venture owners by a book titled "Lean Startup" where the author, Ries, proposed a catalog of ten pivot archetypes [8-10]. Ries' [8] catalog of ten pivot archetypes is not exhaustive and researchers have proposed additional archetypes [9] after the publication of Ries' book. These new pivot archetypes include market zoom-in, complete and side project pivots [10]. Our work is related to this line of research as we also propose a new pivot archetype in this paper – i.e., the Larger Goal pivot.

The Larger Goal pivot represents a situation in which an organization generates new lower-level alternatives (e.g., tactics) to achieve some higher-level objective (e.g., strategy). Casadesus-Masanell and Ricart [2, 3] note that strategy refers to how a firm competes in the marketplace, through its choice of business model, while tactics refer to the residual choices open to a firm by virtue of the business model that it employs. A Larger Goal pivot is necessary in an organization if existing tactical options are inadequate or unsatisfactory for achieving its strategic goals. Larger Goal pivot indicates navigation along a goal hierarchy from existing lower-level goals to higher level-goals and the generation of new lower-level goals from higher-level goals. This approach can be applied to any scenario of business goal change however when a goal hierarchy is involved then it involves Larger Goal rethinking. In this context, the term "Larger" refers only to relative positions of goals in a hierarchy.

In an earlier paper [11], we proposed a goal-modeling based technique using the *i** modeling language for articulating and analyzing pivot archetypes proposed by Ries [8]. In that work [11], we had argued that various types of pivoting follow specific patterns of reasoning. These patterns of reasoning can be abstracted and expressed as conceptual models. We illustrated the application of that technique by instantiating a multi-actor model of a real-world startup in Toronto that undertook pivoting. In that work [11] we proposed strategic patterns and decontextualized representations of Ries' pivot archetypes [8] using the *i** modeling language. For instance, for zoom-in and zoom-out pivots – we needed to represent a hierarchy of needs for narrowing and enlarging the scope of the customer value proposition; and for customer segment pivot – we needed to represent target groups of customers as strategic actors [11]. In [27] we use a retrospective case of Twitter to illustrate the application of conceptual modeling to support pivoting.

In this paper, we propose the Larger Goal pivot as a new type of organizational pivot relative to the archetypes proposed by Ries [8]. We use a retrospective case of Netflix to illustrate the application of conceptual modeling to support pivoting. In a historic case the solution space (i.e., To-Be options) is already known to the modeler. In the real-world, domain specialists and subject matter experts (SMEs) would apply their situational awareness and contextual knowledge to generate a solution space with new alternatives iteratively, creatively, and incrementally.

2 Case Example: Customer Segment Retargeting by Netflix to Achieve Larger Goal

The following summary of this Netflix case is based on published details that were co-authored by the Vice President of Edge Engineering at Netflix in [12]. Netflix operates a streaming video-on-demand platform that allows its subscribers to access its content on a variety of devices including smartphones, tablets, laptops, and desktop computers. It was founded as a postal-mail based DVD rental service in 1997 and transformed into an Internet based video streaming service between 2005 and 2007. Coupled with its international expansion, its transformation contributed to a tenfold growth in Netflix's annual revenues between 2005 and 2016.

A key enabler of Netflix's transformation into a video streaming service was its public Application Programming Interface (API). Netflix had built up an ecosystem of mashup apps over nearly ten years of running a video streaming business. These mashup apps were created by third party developers and combined Netflix assets (e.g., content, catalog) with third Party resources (e.g., forums, feeds) that added value to Netflix services. App developers were either software vendors that created mashups or hardware manufactures that developed device-specific viewer apps.

Netflix cultivated this ecosystem by offering its public API to third party developers because its complementors built synergistic offerings for its subscribers that were outside the core business of Netflix (i.e., video streaming). Examples of such mashups included apps for video recommendations, ratings, rankings, and referrals. Netflix encouraged the proliferation of such mashups because the usage of any mashup necessitated a Netflix subscription which was central to its strategy. Netflix absorbed the costs of maintaining and provisioning its API over time (i.e., to upgrade interfaces, sustain adequate capacity, etc.) as well as of supporting members of its ecosystem (e.g., by updating documentation, performing code reviews, etc.).

In 2014, Netflix decided to shut down its public API and thereby close this ecosystem [13, 14]. Netflix's ecosystem was vibrant at that time however, after being in existence for almost ten years, Netflix's ecosystem had started to return diminishing returns. Specifically, Netflix's approach of growing its revenues from its existing subscribers via its ecosystem stopped contributing substantially to its strategic objective of overall revenue growth. Therefore, Netflix decided to pivot its revenue model to focus on revenue growth from prospective subscribers via its core business to grow its overall revenue. This case example analyzes this pivot that was undertaken by Netflix in 2014 and resulted in the shuttering of its public API.

3. Modeling the Pre-Pivot and Pivot Scenarios

3.1 Pre-Pivot Scenario: Cultivation of Ecosystem via Public API

Following [11] and [27] we use the *i** modeling language to express and analyze pivoting scenarios. We acknowledge that other types of goal modeling languages may also work if they support multiple actors. The *i** language was originally developed to support early stage requirements engineering [15] but has been applied to many other areas involving complex socio-technical phenomena [16] including business model analysis [17], pivoting [11, 27], and strategic coopetition [23-26, 28]. Fig. 1 presents an *i** diagram showing the pre-pivot scenario in the Netflix case study.

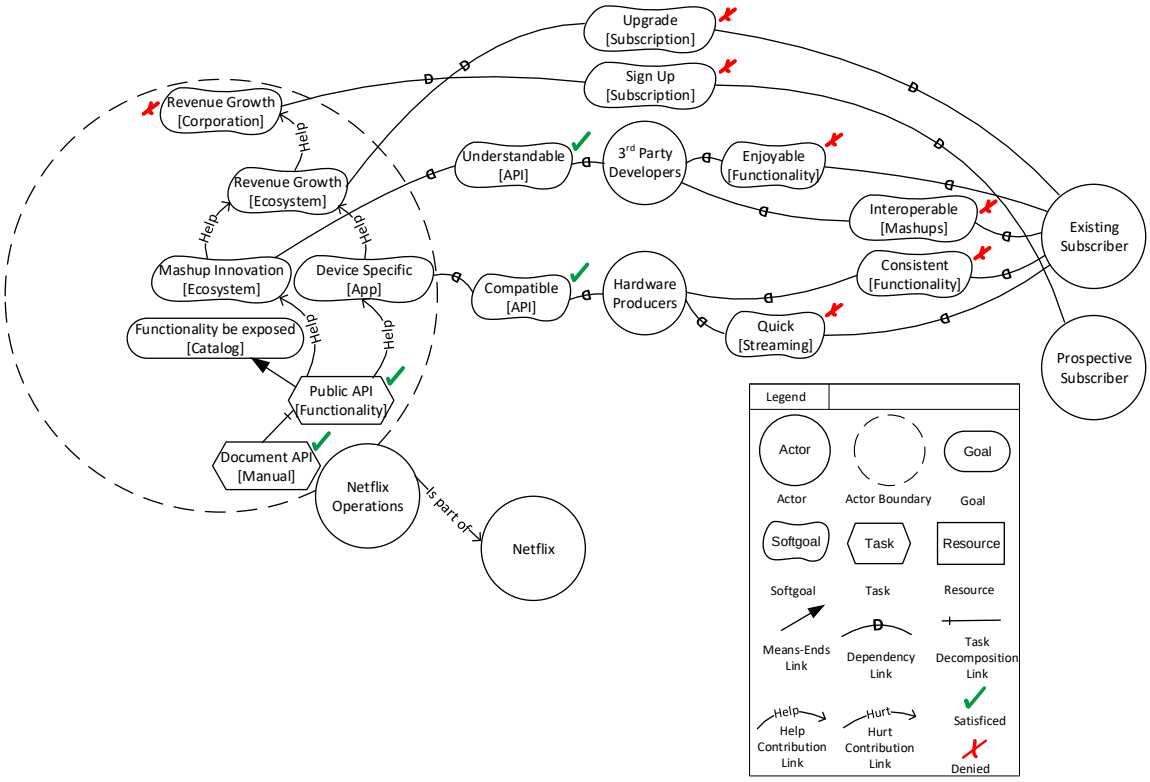

Fig. 1. *i** Strategic Rationale (SR) diagram showing pre-pivot scenario in the Netflix API case

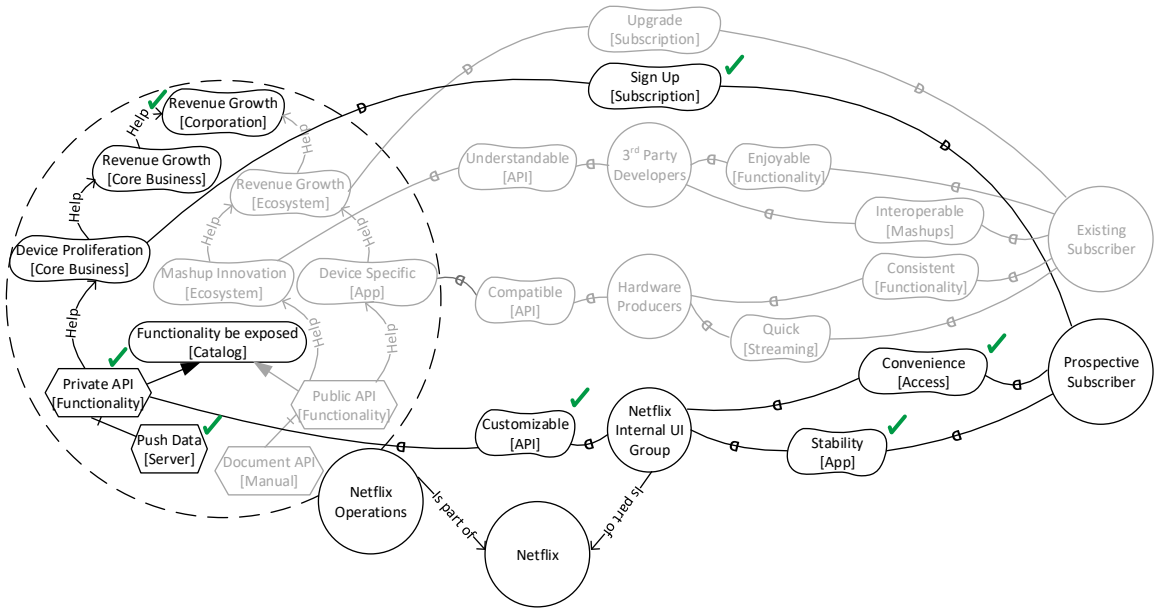

Fig. 2. *i** Strategic Rationale (SR) diagram showing pivot scenario in the Netflix API case

"Netflix Operations" is a business unit within the "Netflix" organization. This is depicted by associating the *actors* "Netflix Operations" and "Netflix" with an *is-part-of* link, which is used to show aggregation. An *actor* is an autonomous, reflective, self-interest seeking, and social agent with a contingent boundary [18]. The primary objective of "Netflix Operations" is "Revenue Growth for the Corporation". This is represented as a *softgoal*, which is a quality objective without clear-cut achievement criteria. Each *actor* seeks to achieve its *softgoals* to a sufficient degree as judged from its own perspective.

"Netflix Operations" can pursue this objective by increasing revenue generated by complementors in its ecosystem. This is depicted by a *Help contribution* link connecting the second-level *softgoal* of "Revenue Growth by the Ecosystem" with the top-level *softgoal* "Revenue Growth for the Corporation". *Contribution* links connect *softgoals* or *tasks* (described below) to other *softgoals* to portray hierarchies of quality objectives and their effects on each other. They are used to denote the positive, negative, neutral, or unknown impact of a *softgoal* or *task* on another *softgoal*.

This aim of increasing revenue generated by complementors in its ecosystem can be achieved by encouraging third party developers to innovate mashups as well as motivating hardware producers to build device specific apps. This is shown by *Help contribution* links linking a higher-level *softgoal* with two lower-level *softgoals* which are: (1) "Mashup Innovation be performed in the Ecosystem by third party developers", and (2) "Device Specific Apps be built by Hardware Producers".

These lower-level *softgoals* are operationalized via a "Public API" that offers the functionality of Netflix to third party developers and hardware producers. This operationalization is portrayed as a *task* which is a means for achieving an end. "Netflix Operations" intends to expose the functionality of its catalog to its complementors. This intention is depicted as a *goal* which is a state of affairs in the world that an *actor* wishes to achieve. Therefore, the *task* "Public API" is connected to the *goal* "Functionality be exposed of the Netflix catalog" via a *means-ends* link.

Means-ends links connect *tasks* to *goals* such that the completion of any *task* leads to the satisfaction of its associated *goal*. A *goal* describes something that should be done while a *task* specifies a particular way in which something should be done.

Netflix must "Document its API" in a manual so that third party developers can use it. This is depicted as a subordinate *task* of the superior *task* "Public API" using a *task-decomposition* link. A *task-decomposition* link connects *tasks* to their subordinate entities which can be *tasks*, *resources*, *goals*, and *softgoals*. Each subordinate entity of a *task* must be accomplished for that *task* to be completed. Therefore, *means-ends* links are treated as logical OR while *task-decomposition* links are treated as AND when evaluating goal achievement.

Actors in *i** may depend on other actors for *goals* to be achieved, *tasks* to be completed, *resources* (i.e., a physical or informational entity) to be obtained, and *softgoals* to be accomplished. For example, "third party developers" depend on "Netflix Operations" for an "Understandable API" while "Hardware Producers" depend on "Netflix Operations" for a "Compatible API".

An *actor* that depends on another *actor* is referred to as a depender while the *actor* on which the depender depends is referred to as a dependee. The depender de-

pends on the dependee for a dependum. While a dependency can be beneficial for a depender it can also be deleterious since any dependum can make a depender vulnerable to exploitation and opportunism by its dependee. The curved side of the character 'D' in the *Dependency* link points towards the dependee while the flat side points towards the depender.

In the Netflix case, "Existing Customers" of Netflix depend on "third party developers" for mashups that are "Enjoyable" as well as "Interoperable" with each other and they also depend on "Hardware Producers" for device specific apps that offer "Quick Streaming" as well as "Consistent Functionality". "Netflix Operations" depends on "Existing Subscribers" to "Upgrade" their Subscriptions due to the beneficial value propositions of mashups by "third party developers" as well as device specific apps by "Hardware Producers".

After a model has been developed it can be used to assess the viability and desirability of alternative means for achieving an end. The goal graph is crucial for performing trade-off analysis in *i** models. A technique for forward propagation of *contribution* links is described in [19]. In this technique, propagation rules are applied to attach current values (i.e., satisfied, denied, etc.) from offspring to their parents and the resolution of the *softgoal* labels is performed at the parent level [20]. Viability of a particular *task* is evaluated by checking whether it satisfies or denies certain *softgoals*. The selection of an unviable alternative at a lower-level can lead to the denial of an important objective at the higher-level.

Alternative means (i.e., *tasks*) for achieving an end (i.e., *goal*) can be compared on the basis of the impact of each *task* on relevant quality objectives (i.e., *softgoals*). Desirability of a particular *task* is examined by comparing the *softgoals* that are satisfied or denied by that *task* with the *softgoals* that are satisfied or denied by other *tasks*. The selection of an undesirable alternative at the lower-level means that better alternatives for achieving an objective at the higher-level are not selected.

Forward propagating satisfaction labels via *contribution* links reveals that "Netflix Operations" published an API that was "Understandable" by "third party developers" and "Compatible" for "Hardware Producers". These dependencies are denoted with ✓. Nonetheless, "third party developers" were unable to offer mashups to "Existing Subscribers" of Netflix that were "Enjoyable" or "Interoperable". Similarly, "Hardware Producers" were unable to offer device specific apps to "Existing Subscribers" of Netflix that supported "Quick Streaming" or "Consistent Functionality". As a result, "Existing Subscribers" of Netflix did not "Upgrade" their Subscriptions.

This led to the denial of the Larger Goal for "Netflix Operations" which was "Revenue Growth for the Corporation". Therefore, each of these dependencies are denoted with ✗. This means that "Netflix Operations" was bearing the cost of supporting a public API for its partners but was not benefiting from that public API in terms of substantial contributions to its strategic objective.

Subsequently, "Netflix Operations" decided to pivot away from its approach of "Revenue Growth by the Ecosystem" to achieve its Larger Goal of "Revenue Growth for the Corporation". It switched to an approach of "Revenue Growth from its Core Business" to achieve its Larger Goal of "Revenue Growth for the Corporation". This pivot is discussed in the next sub-section.

3.2 Pivot Scenario: Service Proliferation on Devices via Private API

The first step of the Larger Goal pivot of “Netflix Operations” starts with identifying the highest level strategic objective that it needs to achieve. This is done by tracing the links from the pre-pivot low-level operationalization (i.e., *task*) upwards to the highest-level objective (i.e., *softgoal*). The operationalization that "Netflix Operations" was pivoting away from entailed offering a “Public API” and the highest level strategic objective that this operationalization was related to was “Revenue Growth for the Corporation”. This strategic objective was not satisfied via the low-level operationalization of offering a “Public API”.

Therefore, in the second step of the Larger Goal pivot, “Netflix Operations” needs to create a new way to satisfy this strategic objective. Domain Specialists and Subject Matter Experts (SMEs) in “Netflix Operations” decided to abandon the approach of “Revenue Growth by the Ecosystem” since it was related to the low-level operationalization that entailed offering a “Public API”. Instead they adopted the approach of “Revenue Growth from its Core Business” which entailed shifting the revenue growth focus away from its “Existing Subscribers” and onto its "Prospective Customers". This shift represents a Customer Segment pivot per the pivot archetypes of Ries [8].

The pre-pivot scenario lacked an operationalization for encouraging “Prospective Subscribers” to “Sign Up” for new Subscriptions. Therefore, in fig. 1, the dependum “Sign Up” for new Subscriptions is connected to the highest level strategic objective of “Netflix Operations”. In the third step of the Larger Goal pivot, SMEs in “Netflix Operations” designed and explored new alternatives for satisfying the strategic objective in a systematic and structured manner. This step extended the goal graph from the pre-pivot scenario to include new model elements in the pivot scenario. The pivot scenario is depicted in fig. 2. For ease of interpretation in the visual presentation of fig. 2, existing model elements from fig. 1 are greyed-out and new model elements are depicted in black color.

In the pivot scenario, the highest-level objective of “Revenue Growth from Core Business” is refined into a new approach of “Device Proliferation”. This lower-level aim entailed the creation of a standardized app for watching videos on Netflix that works across a wide range of device families (not shown*). A standardized app offers consistent features as well as uniform functionality across device families (not shown*). Moreover, it is less costly to build and maintain a single app that is stable than many apps that are stable (not shown*).

* In this instance, and in the remainder of this paper, certain aspects of the relationship between *actors* are not shown due to page limitations.

In the pivot scenario, "Prospective Customers" depended on Netflix for a "Stable App" that afforded them "Convenient Access" to the Netflix catalog and content. "Netflix Operations" depended on "Prospective Customers" to "Sign Up" for new Subscriptions. However, "Netflix Operations" was not experienced in designing user interfaces (UIs). In the pre-pivot scenario, “third party developers” and “Hardware Producers” designed mashups and apps for watching Netflix videos.

In the pivot scenario, “Netflix Operations” needed to find a different way to build a standardized app for watching videos on Netflix. For this purpose, “Netflix Operations” established the “Netflix Internal UI Group” which was comprised of staff members on the Netflix payroll. The “Netflix Internal UI Group” depended on “Netflix Operations” for a “Customizable API”. Since the “Netflix Internal UI Group” was a part of “Netflix” then “Netflix Operations” only needed to offer a “Private API” to it. “Netflix Internal UI Group” could leverage a “Customizable” “Private API” to build a standardized app for watching Netflix videos. “Netflix Operations” merely needed to “Push Data” onto a Server that was accessible to “Netflix Internal UI Group” via this “Private API”.

“Netflix Internal UI Group” used this “Private API” to design and distribute a “Stable App” to “Prospective Subscribers”. These “Prospective Subscribers” were able to use this app to “Conveniently Access” Netflix services. This incentivized “Prospective Subscribers” to “Sign Up” for a Netflix subscription and helped “Netflix Operations” to achieve its aim of “Device Proliferation”. Consequently, “Device Proliferation” allowed “Netflix Operations” to satisfy its higher-level objective of “Revenue Growth for the Core Business” and ultimately satisfy its highest-level objective of “Revenue Growth for the Corporation”.

4 Related Work

This paper contributes to the body of research literature pertaining to Enterprise Modeling (EM) of organizational pivots. Currently, EM research that is exclusively focused on pivoting in organizations is relatively scarce. However, the body of research literature on EM of organizational strategy (of which pivoting is one part) is comparatively richer. We [11] adopt *i** to model various types of pivots in startups and large enterprises. We also [27] present conceptual models of pivoting based on a retrospective case example of Twitter. Giannoulis et al. [21] offer a language for modeling strategy maps. Kim et al. [22] propose a modeling technique to depict a value chain of a virtual enterprise. We introduced a technique for modeling and analyzing strategic coopetition between organizations [23, 24] as well as its characteristics of complementarity [25] and reciprocity [26].

5 Conclusion and Future Work

We utilized a strategic modeling approach to systematically search for and create viable approaches for implementing a Larger Goal pivot. The approach available in the pre-pivot scenario was shown to be inadequate for meeting the strategic objective of the focal organization. Therefore, a pivot scenario was generated that encompassed the design of a new approach for meeting the Larger Goal of the focal organization. An abstract pattern and decontextualized representation of Larger Goal pivot has been

developed and future work includes validating this model in real world organizational settings. Future work also includes developing a catalog of pivoting goals to serve as a knowledge base for SMEs and domain specialists.

Future work also seeks to address certain limitations of *i** modeling that were encountered during the expression and analysis of the Netflix case. *i** models have limited visual scalability in terms of human interpretability. Goal graphs with multiple actors and multiple goal structures can become inscrutable for humans. *i** models do not support the depiction of temporality and therefore pre-pivot and pivot configurations are depicted in separate diagrams. This requires a model analyst to switch back and forth between the models to compare them. *i** models lack support for depiction of negative dependencies and therefore it is not possible to perform counterfactual reasoning.

Some of these limitations can be partially addressed with tool support. A tool for *i** modeling can help to make *i** models more explainable to humans. Features and functions of such a tool might include expanding/collapsing, revealing/hiding, enlarging/shrinking, and coloring/discoloring parts of the *i** model. A tool for *i** modeling can also help with model evaluation by calculating satisfaction of goals in a model. It can do so by propagating satisfaction labels across elements over contribution links and then applying rules to resolve a single label for each goal from contributions to it.

References

1. Ghemawat, P. (2002). Competition and Business Strategy in Historical Perspective. Business History Review, 76(1), 37-74.
2. Casadesus-Masanell, R., & Ricart, J. E. (2010). From Strategy to Business Models and onto Tactics. Long Range Planning, 43(2-3), 195-215.
3. Casadesus-Masanell, R., & Ricart, J. E. (2011). How to Design a Winning Business Model. Harvard Business Review, 89(1/2), 100-107.
4. Stanton, P. (2016, August 30). Microsoft's Pivot and the Importance of Windows Containers. Retrieved from https://www.infoworld.com/article/3113161/it-management/microsofts-pivot-and-the-importance-of-windows-containers.html.
5. Townsend, K. (2016, July 15). Why Microsoft's Linux Lovefest Goes Hand-In-Hand with its Azure Cloud Strategy. Retrieved from https://www.techrepublic.com/article/why-microsofts-linux-lovefest-goes-hand-in-hand-with-its-azure-cloud-strategy/.
6. Baribeau, S. (2013, January 24). Pivoting from Auctioneers to Online Sellers–eBay Takes on Amazon. Retrieved from https://www.fastcompany.com/3004125/pivoting-auctioneers-online-sellers-ebay-takes-amazon.
7. Weissbrot, A. (2017, November 10). Inside eBay's Repositioning as a Modern Ecommerce Platform. Retrieved from https://adexchanger.com/advertiser/inside-ebays-repositioning-modern-ecommerce-platform/.
8. Ries, E. (2011). The Lean Startup: How today's entrepreneurs use continuous innovation to create radically successful businesses. New York: Crown Publishing Group.
9. Edison, H., Smørsgård, N. M., Wang, X., & Abrahamsson, P. (2018). Lean Internal Startups for Software Product Innovation in Large Companies: Enablers and Inhibitors. Journal of Systems and Software, 135, 69-87.
10. Bajwa, S. S., Wang, X., Duc, A. N., & Abrahamsson, P. (2017). "Failures" to be Celebrated: An Analysis of Major Pivots of Software Startups. Empirical Software Engineering, 22(5), 2373-2408.

11. Pant, V., Yu, E., & Tai, A. (2017, November). Towards Reasoning About Pivoting in Startups and Large Enterprises with i*. In IFIP Working Conference on The Practice of Enterprise Modeling (pp. 203-220). Springer, Cham.
12. Jacobson, D., Woods, D., & Brail, G. (2011). APIs: A Strategy Guide. O'Reilly Media, Inc.
13. Jacobson, D. (2014, January 14). Why you Probably Don't Need an API Strategy. Retrieved from https://thenextweb.com/entrepreneur/2013/09/15/why-you-probably-dont-need-an-api-strategy/.
14. Lawler, R. (2014, June 13). Netflix Will Shut Down Public API Support for Third-Party Developers on November 14. Retrieved from https://techcrunch.com/2014/06/13/netflix-api-shutdown/.
15. Yu, E. S. (1997, January). Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering. In Proceedings of the Third IEEE International Symposium on Requirements Engineering, pp. 226-235. IEEE.
16. Yu, E., Giorgini, P., Maiden, N., & Mylopoulos, J. (2011). Social Modeling for Requirements Engineering. MIT Press.
17. Samavi, R., Yu, E., & Topaloglou, T. (2009). Strategic Reasoning About Business Models: A Conceptual Modeling Approach. Information Systems and e-Business Management, 7(2), 171-198.
18. Yu, E. (2001). Agent Orientation as a Modelling Paradigm. Wirtschaftsinformatik, 43(2), 123-132.
19. Horkoff, J., & Yu, E. (2013). Comparison and Evaluation of Goal-Oriented Satisfaction Analysis Techniques. Requirements Engineering, 18(3), 199-222.
20. Horkoff, J., & Yu, E. (2011, March). Analyzing Goal Models: Different Approaches and How to Choose Among Them. In Proceedings of the 2011 ACM Symposium on Applied Computing (pp. 675-682).
21. Giannoulis, C., Petit, M., & Zdravkovic, J. (2010, November). Towards a Unified Business Strategy Language: A Meta-Model of Strategy Maps. In IFIP Working Conference on The Practice of Enterprise Modeling (pp. 205-216). Springer, Berlin, Heidelberg.
22. Kim, C. H., Son, Y. J., Kim, T. Y., Kim, K., & Baik, K. (2006). A Modeling Approach for Designing a Value Chain of Virtual Enterprise. The International Journal of Advanced Manufacturing Technology, 28(9-10), 1025-1030.
23. Pant, V., & Yu, E. (2018). Modeling Simultaneous Cooperation and Competition Among Enterprises. Business & Information Systems Engineering, 60(1), 39-54.
24. Pant, V., & Yu, E. (2016). Coopetition with Frenemies: Towards Modeling of Simultaneous Cooperation and Competition among Enterprises. In IFIP Working Conference on The Practice of Enterprise Modeling (pp. 164-178). Springer, Cham.
25. Pant, V., & Yu, E. (2017) Modeling Strategic Complementarity and Synergistic Value Creation in Coopetitive Relationships. In International Conference of Software Business (pp. 82-98). Springer, Cham.
26. Pant, V., & Yu, E. (2018) Generating Win-Win Strategies for Software Businesses under Coopetition: A Strategic Modeling Approach. In International Conference of Software Business. Springer, Cham.
27. Pant, V., & Yu, E. (2018) Conceptual Modeling to Support Pivoting – An Example from Twitter. In Proceedings on Advances in Conceptual Modeling - ER 2018 Workshops AHA, MoBiD, MREBA, OntoCom, and QMMQ, Lecture Notes in Computer Science. Springer, Verlag.
28. Pant, V., & Yu, E. (2018). Getting to Win-Win in Industrial Collaboration under Coopetition: A Strategic Modeling Approach. In International Conference on Business Informatics Research. Springer, Cham.