The Antecedents of Transaction Costs in Digital Ecosystems: A Configurational View on the Interplay of App Architecture and Platform Governance

Dominik Dellermann
University of Kassel
Dellermann@wi-kassel.de

Fabian Reck
University of Bamberg
Fabian.reck@uni-bamberg.de

Abstract

The locus of value creation and innovation in the software industry is shifting more and more to platform ecosystems on which numerous developers create extensions with additional functionalities based on the platforms core architecture. While such complementors may strongly profit from platforms, there are considerable costs. Recent studies therefore examined the costs of fitting apps to the specifications of certain platforms; however, these works largely neglect costs arising from the transactional relationship between platform and complementor. In order to shed light on this, our work examines how design choices of platform governance and app architecture impact the emergence of four types of cost-inducing hazards within the transactional context of the ecosystem. By using a configurational approach based on fuzzy-set qualitative comparative analysis (FsQCA), we display complex interactional effects of the causal conditions on complementors' perception of hazardous environments and thus provide valuable insights for both practice and theory on platform ecosystems.

1. Introduction

The emergence of digital platforms like Salesforce´s Force.com or SAP´s HANA substantially changed the logic of innovation in the software industry. Contrary to traditional approaches of software engineering, modern software development strongly relies on innovation from third-party developers, called complementors [15]. The platform owner (e.g SAP) provides an expandable resource base, which enables other actors to develop applications that extend the basic functionality of the platform. As a result, an ecosystem emerges around the platform in which platform owners and third-party developers interact in complex ways. In such systems, the innovative performance of single firms is not only determined by the innovative capabilities of a single firm, but considerably influenced by the properties of the surrounding ecosystem [45].

An essential task of a platform owner is therefore to attract complementors to join the ecosystem and thereby facilitate innovation and the generation of complementary value propositions [5]. The predominant current recipe to achieve this is by creating and offering motivational factors and relational rents [8, 19]. However, famous examples like Blackberry´s mobile operating system show that gaining solid traction among third-party developers remains challenging. Often, platform ecosystems are suffering fluctuation and high rates of desertion [35, 36].

A currently under-researched explanation for this low traction and high rates of fluctuation may lie in the fact that complementors face considerable costs when joining a specific platform. If these costs are too high, they can rapidly outweigh the additional value generated by the ecosystem and provoke complementors' abandonment of a platform [44]. Previous studies that addressed this question primarily took a technical perspective and focused solely on the coordination costs related to platform dependencies [35, 36]. However, to provide a more holistic analysis of costs related to development activity on a platform, also economic dimensions need to be considered. In this context, transaction cost theory (TCT) is one of the most prominent theoretical bases to explain and predict relationships and boundary decisions associated with interfirm exchange [2, 41]. From the perspective of TCT, joining a platform might induce a cost disadvantage relative to vertically integrated structures. These costs can take different forms (e.g. search and information costs; investments in social relations; opportunity costs) which are mainly determined by four potential hazards: behavioral uncertainty concerning the platform owner, technological and market uncertainty as well as asset specificity [29]. To understand fluctuation in platform ecosystems, it is therefore beneficial to take one step back and examine

URI: http://hdl.handle.net/10125/41800
ISBN: 978-0-9981331-0-2

what causes the emergence of these four potential hazards from the perspective of an individual complementor.

Within this paper, we develop the overarching idea that specificity, behavioral, technological and market uncertainty are mainly influenced by design choices taken by the platform owner as well as the individual third-party developer. In particular, the interplay of the control modes applied to govern the surrounding ecosystem as well as the micro-architecture of single apps will affect the extent to which the developer faces cost-inducing hazards. As past research highlighted the complex interplay of platform governance and app micro architecture [35, 36], we decided not to rely on regression-based net effect models but to apply configurational theory and qualitative comparative analysis (QCA) to this end. In doing so, we are able to capture complex combinatorial causal effects *"in which an outcome may follow from several different combinations of causal conditions"* ([28]: p. 23). This point of view is grounded in configurational theory and allows us to apply a different paradigmatic lens for understanding the complexities of platforms and ecosystems [10]. In all, we aim at answering the following research question:

Which configurations of architectural choices and platform governance mechanisms minimize asset specificity, behavioral uncertainty, technological and market uncertainty in platform ecosystems?

2. Conceptual background

2.1. Transaction costs from a third-party developer's perspective

Transaction cost theory (TCT) [42, 43] provides indications that by entering partnerships third-party development might induce a cost disadvantage relative to vertically integrated structures of software development [44]. Such costs can arise due to certain patterns of human behavior and characteristics of the transaction environment [42, 43]. As human decision making is characterized by bounded rationality, i.e. an individual´s limitation in information processing and rational decision making due to constraints on their cognitive capabilities, occasions in which opportunistic behavior might be favorable naturally emerge [42]. Turning to the context of platform ecosystems, such potential hidden agendas bear considerable risk for complementors, especially as the development on a certain platform oftentimes involves high amounts of up-front investments or commitments [44]. For instance, knowledge leakage and imitation of own software solutions may threaten the complementor's ability to create appropriable rents from these investments [24]. As a consequence, third-party developers will make further investments in order to decrease such hazards, e.g. by acquiring intellectual property rights and legal support [8]. The likelihood and amount of potential losses due to opportunistic behavior in the ecosystem is to a large part determined by the transaction environment [42]. Therefore, the amount of transaction costs that a complementor is willing to accept mirrors the perceived degrees of the four main determinants of transaction costs, namely asset specificity, behavioral uncertainty on behalf of the partner, as well as technological and market uncertainty [29].

Asset specificity: The concept of asset specificity describes the degree to which an investment will lose value if the resource must be adopted for other purposes. Specific assets are significantly more valuable in a particular exchange relationship than within alternative partner relations and lead to a 'lock-in' effect to a certain platform [41]. If a complementor has to spend significant amounts of time and money in order to create tailored procedures, routines and systems concerning one particular platform, these "sunk" investments are likely to be of no value in the face of migrating to another platform [46]. Consequently, if a complementor perceives high levels of asset specificity it will expect high transaction costs.

Behavioral uncertainty: Second, behavioral uncertainty arises from the instance that partnership evaluation is often complex and the partner's actions and performance are hard to evaluate [29]. This can be further strengthened if the partner refuses to share information, disguises or distorts it [34]. Particularly within the context of platforms, there are quite asymmetric relationships between platform owners and complementors [7] so that a large part of the costs emerging from behavioral uncertainty is on the complementors' side. So, the less predictable the platform owner's behaviors are, the more complementors will perceive the necessity to invest in corresponding safeguards.

Technological uncertainty: Third, technological uncertainty as one key dimensions of environmental turbulence refers to the unpredictability of the firm's environment. While technological evolution is unpredictable in principle [37], complementors furthermore face technological uncertainties especially because it is the platform owner who sets crucial technological framing conditions like for instance APIs, SDKs, system governance (component boundaries and real-time support) and shared assets (e.g., maps, fields for data input-output) [6]. These may heavily influence the value and functionality of new and existing apps. Third-party developers therefore face an adaptation problem and might be forced to adjust internal resources, external agreements and especially the relationship towards the platform owner in order to fit the new external circumstances [29]. Hence, on the one hand, the less predictable the technological surroundings are, the more likely are up-front investments to make agreements adaptive. On the other hand, uncertainty may induce opportunistic behavior by the platform owner, e.g. through extracting concessions at the partner's expense [40]. The consequence of this is a propensity of higher transaction costs.

Market uncertainty: Volatility of market conditions (e.g. market, demand, and competitive environment) is the second dimension of environmental uncertainty. This volatility represents a crucial source of cost for complementors, as for instance the sustainability of their specific niche is required for them to succeed [26]. In the context of platforms, the degree of market uncertainty quite strongly depends on the characteristics of the platform. As the platform mediates the relationship between end users, the complementor and its peers, platform design choices such as licensing policies or the provision of technological support may considerably influence the volatility in platform entrance, desertion or competitive behavior [9]. Facing these obstacles, complementors are confronted with another type of adaptation problem which analogously to technological uncertainty might induce higher costs.

2.2. Platform governance and app microarchitecture as configurational antecedents of cost-inducing hazards

Fig. 1 illustrates the framework of this research. It proposes that the configuration of platform governance mechanisms and app microarchitecture influences a complementor´s transaction costs in platform ecosystems [37]. Rather than relying on conventional reductionistic models, we assume that a specific outcome depends on the configuration of variables rather than the net effects of the individual factors. This approach is suitable for the context of our research for two reasons.

On the one hand, configurational approaches treat sets of variables as single predictors [25]. These serve to simultaneously explain the outcome of interest so that configurational theory is well suited to uncover synergetic and complementary effects [28]. This resonates well with current theoretical perspectives on platform and ecosystem management. These highlight the inseparability of platform governance from app architecture in their mutual effect on organizational outcomes. For instance, Tiwana [35, 36] shows that the evolutionary capabilities of a platform are catalyzed by the level of input control because app modularization by itself is not sufficient to accelerate evolution as every change of the extension requires guaranteeing interoperability and quality standards of the ecosystem. Each variable in isolation therefore has no robust effect on this outcome.

On the other hand, configurational theories assume asymmetric rather than symmetric relations between conditional and outcomes variables [10]. This implies equifinality between different initial conditions and development paths [28]. Consequently, conditions may either be sufficient or necessary causes of a dependent variable. According to both organizational [12] and information systems research [23] such notions superiorly accommodate organizational realities. As El-Sawy et al. [10] point out, this perspective therefore particularly fits the purpose of analyzing the organization of digital ecosystems.

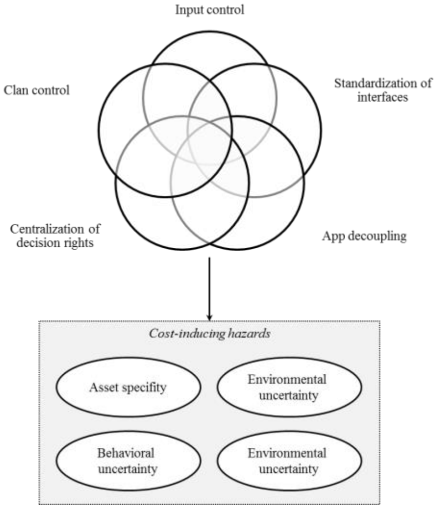

Input control
Clan control
Standardization of interfaces
Centralization of decision rights
App decoupling
Cost-inducing hazards
Asset specifity
Environmental uncertainty
Behavioral uncertainty
Environmental uncertainty

Figure 1. Research framework

In the selection of our causal conditions, we follow notions of Tiwana et al. [37] and Tiwana [35] on IT and especially intraplatform dynamics. The former outline three core mechanisms of how to execute IS governance: decision rights, control and system architecture. Building on this foundation, Tiwana [35] outlines the inseparability of these mechanisms as predictors of platform outcomes. In line with previous work that focuses on the influence of governance on transaction costs, we therefore rely on configurations of these three governance dimensions to explain cost-inducing hazards [2]. We operationalize platform governance as both the control mechanisms and the locus of decision rights within the relationships between platform owner and the individual third-party developers. In the context of platforms, such governance mechanisms are usually designed and introduced by the platform owner and are not necessarily suited to reduce costs bilaterally. While certain mechanisms may allow the platform owner to minimize asset specificity and uncertainty, some of those may simultaneously increase the transaction costs of complementors. Still, also third-party developers possess design alternatives based on which they can influence their relation to the platform. Concretely, the microarchitecture (in contrast to the macro architecture of the overall platform) of their apps allows complementors to minimize costs by exploiting benefits of modularization of the linkages between the platform and the app [35, 36]. Summing up these insights, our set of causal conditions encompasses the most commonly applied IS governance forms on platforms. It includes design elements from control (clan control, input control), decision rights partitioning (centralization of decision rights) as well as architecture (app decoupling and standardized interfaces).

The first dimension, control, refers to the mechanisms established by a platform owner that govern actions of the partners. Decision rights as the second dimension of platform governance encompass extension app features, functionality, design, and implementation procedures [34, 35]. Both elements represent design choices which are largely on the behalf of the platform owner.

Clan control: The most common informal mechanism to govern partners and the interaction within an ecosystem is clan control. This form of control is accomplished by mutual values and shared goals between the "clan" of complementors and the platform owner [17]. For instance, platform owners may release norms, mutual values and goals that are beneficial for the platform. When either a complementor or the platform owner shows deviant behavior, other members of the clan might react with social sanctioning. On the dyadic level, clan control therefore results in behaviors of both the platform owner and the complementor that would not violate such values by acting opportunistically. As a result, levels of uncertainty for the complementor will be reduced if clan control is prevalent within the platform ecosystem.

Input control: A suitable formal control mechanism for platform ecosystems is input control. It describes the degree to which platform owners control complementary apps by utilizing application and selection processes [35]. Hence, not all complementary apps are admitted to the ecosystem. Input control keeps tabs on the admission to the ecosystem and allows the platform owner to guarantee interoperability, quality or the fit with the platform's interests, values, and positioning [36].

Centralization of decision rights: Apart from different modes of control, another central element of platform governance is the degree to which decision rights are centralized or delegated [36]. This form of governance encompasses different classes of decision rights [11]. Taken to the platform context, platform owners distribute decisions about what an app should do (e.g., features and functionality), how it should do it (e.g., design, user interface), and the control of boundary resources (e.g. the platform´s interfaces) among itself and the complementors [36]. Though platform owners are often willing to delegate decision rights to complementors because these possess nuanced knowledge about the app's means and ends, in the case of strategically relevant extensions, owners may decide to keep or retract that authority [35, 36].

While the mode of platform governance represents a design choice on the behalf of the platform owner, the individual complementors may choose corresponding design elements on their own [35, 36]. Thereby, app modularization is focusing on linkages between the platform and the app, which is defined as the "degree to which an app is loosely coupled and interacts through standardized interfaces with the platform." ([34]: p. 268). In general, modularity reduces the complementor's specific investments in knowledge and technologies that are bound to a certain platform and thus, in line with previous literature on modularity [3], we thus assume that modularization reduces transaction costs. The architectural concept of modularization can be theoretically described along two distinctive dimensions: decoupling of an app from the platform, and the use standardized interfaces as linkage [31].

App decoupling: The level of decoupling describes an architecture in which changes within the architecture of the platform do not have any ripple effect on the single app. The more decoupled an app is, the more independently it can be developed by a complementor while still ensuring fluent interoperation with the platform. Usually, the complementor makes such a design choice within the exogenous constraints of the platform and minimizes the platform dependencies on the minimal degree to which an app is required to be conforming to the specifications interface [35]. This is achieved by carefully selecting and placing "thin connections" between app and platform while removing the remaining ones so that changes to the app or the platform do not condition changes to the respective counterpart [39].

Standardization of interfaces: On the other hand, the standardization of interfaces describes the degree to which the linkages between the single app and the platform are stable, formalized and well-documented [36]. Thereby, stability is ensured by the existence of boundary resources like application programming interfaces (APIs) [36]. Such standards codify the relationships between the app and the platform as well as clearly articulate rules and specifications for apps and platform infrastructure. Such clarity and transparency might help to overcome issues of opportunism and bounded rationality, so that transaction costs can be reduced.

3. Research methodology

3.1. Data collection

The sampling frame of our research consists of 750 firms equally distributed among the complementors of five leading cloud platforms (i.e. Microsoft Azure, Oracle Cloud Platform, Amazon Web Services, SAP HANA, and Salesforce Force.com). The platforms were chosen for two reasons. First, they are all well-established and have a solid traction among complementors. Second, due to their size and high level of power imbalance, they perfectly meet our requirements for analyzing asymmetric third-party relationship and the corresponding risk.

Congruent with previous surveys of third-party innovators [4], we utilized a web-crawling approach which randomly collected contact data from the platforms´ app stores. A link to the online questionnaire was sent via mail and recipients were asked to forward the questionnaire to high-level executives (C-level; IT executives) as key informants [20]. The invitation mail and the start page of our survey included the purpose of the study and ensured confidentiality and anonymity to the participants.

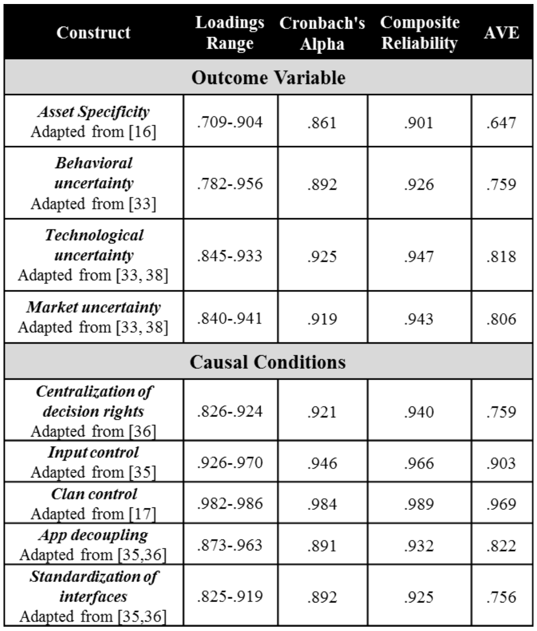
Our sampling approach resulted in a total of N=42 valid cases (response rate: 5.6 percent), which is a common response rate in such settings. We assessed non-response bias by comparing response of early and late respondents [1]. T-tests between the means of the early and late respondents did not reveal any significant differences (p > 0.05), hence rejecting the presence of non-response bias in our study.

Complementors in our study were distributed among all five platforms (Microsoft Azure: 9; Oracle Cloud Platform: 4; Amazon Web Services: 2; SAP HANA: 9; and Salesforce Force.com: 14). Most of our respondents were high-level executives (C-level: 71.4 percent; BU executives: 19 percent). Participants in our sample indicated that they are highly experienced in this topic (>10 years: 83.3 percent) and were experts in the context of our survey (95.2 percent).

3.2. Measurement validation

We used a pilot study with managers in the software industry to construct our measurement instrument and ensure validity, reliability as well as rigor of our main study's results [22]. Therefore, we adapted existing scales to the platform context. Refined items were again evaluated in a pre-test to ascertain that our survey items were interpreted unambiguously (see Table 1).

Table 1 Measurement scales



Construct	Loadings Range	Cronbach's Alpha	Composite Reliability	AVE
Outcome Variable				
Asset Specificity Adapted from [16]	.709-.904	.861	.901	.647
Behavioral uncertainty Adapted from [33]	.782-.956	.892	.926	.759
Technological uncertainty Adapted from [33, 38]	.845-.933	.925	.947	.818
Market uncertainty Adapted from [33, 38]	.840-.941	.919	.943	.806
Causal Conditions				
Centralization of decision rights Adapted from [36]	.826-.924	.921	.940	.759
Input control Adapted from [35]	.926-.970	.946	.966	.903
Clan control Adapted from [17]	.982-.986	.984	.989	.969
App decoupling Adapted from [35,36]	.873-.963	.891	.932	.822
Standardization of interfaces Adapted from [35,36]	.825-.919	.892	.925	.756

Our constructs display evidence of reliability with Cronbach's α greater than .85 for all variables. Confirmatory factor analysis yielded adequately high factor loadings concerning the latent constructs and allows us to assert discriminant validity as the Fornell Larcker criterion is fulfilled for all our study variables [13]. To test common method bias, we conducted Harman's one-factor test [27]. The unrotated factor solution resulted in 8 factors explaining 85 percent of the variance (33 percent was the largest variance explained by one factor). Thus, common method bias is unlikely to be a problem.

3.3. Fuzzy-set QCA

We analyzed the obtained data via Fuzzy-set QCA (FsQCA). FsQCA draws on a set-theoretic approach which emphasizes the effects of the whole rather than its pieces. Hence, it explicitly acknowledges the multidimensionality of research cases [12]. Thereby, FsQCA evaluates the predictive power of the potentially possible configurations of study variables based on measures of consistency and coverage. Consistency values are analogous to correlation estimates in statistical methods. This indicator displays to which degree cases that share a certain combination of conditions agree in leading to a given outcome [28]. Coverage on the other hand represents the degree to which a configuration accounts for the instances of an outcome. Thus, coverage values are analogous to R-square values in regression analysis. For a single configuration, FsQCA calculates two different coverage values. The raw coverage provides a measure which displays in what percentage of the cases in the dataset this particular configuration can be observed [28]. As a single case may be covered by different configurations, it is useful to additionally calculate unique coverage. This indicator estimates the percentage of cases that display membership in the configuration of interest, but not in any other configuration in the solution set [28]. In order to detect configurations which are consistent to the outcome in question, FsQCA comprises three steps [28]: calibration, construction of truth tables, truth table analysis.

The first step is calibration. As FsQCA construes configurations of conditional variables as sets in which research cases can be either member or non-member, calibration of the variable measures is necessary in order to obtain the cases' degree of memberships in a certain set of conditions (here, e.g. membership in the group of firms with highly decoupled apps). Thus, all construct measures must be transformed from Likert scale values into fuzzy set membership scores which range between 0 and 1 with 0 indicating full non-membership, 1 indicating full membership and 0.5 representing the crossover point [32]. Within this work, we choose the observed maximum and minimum values within the sample to specify full membership and full non-membership for all variables and the calculated scale midpoint (median of observed values) as cross-over point. This is analogous to the calibration approach by Fiss [12]. The three values gained through this procedure served as input variables for the calibration procedure in the FsQCA software program [28] which transforms all Likert scale measures to set membership scores.

The construction and refinement of truth tables represent the second step of analysis. A truth table is a matrix of all possible configurations of antecedent conditions. In our case, it consists of 32 rows (in general 2^k, where k is the number of conditions observed [28]). Subsequently, the truth table is refined so that it fits the requirements of FsQCA. In the refinement procedure, each possible configuration is assessed based on two criteria: frequency and consistency. The frequency criterion captures how many empirical cases with a membership score of more than 0.5 in a certain configuration are necessary in order to include the configuration in further analysis. The standard threshold for frequency in medium-sized samples is 1 meaning that every configuration that exists in the empirical dataset will be part of the analysis [32]. The consistency criterion captures if a truth table row displays a consistent subset of the outcome and should outreach a value of at least .8 [28]. In this work, we choose a rather conservative threshold of .9. Overall, 32 cases involved configurations exceeding the frequency threshold of which 7 also exceeded the consistency threshold for asset specificity, 17 for behavioral uncertainty, 18 for technological uncertainty and 17 for market uncertainty.

Finally, in the third step, the truth tables are analyzed based on Boolean algebra. Counterfactual analysis represents the basic principle of this step. The approach applies the Quine-McCluskey algorithm that identifies combinations of factors which consistently lead to a certain outcome by stripping away factors that are inconsistently present or absent concerning the particular outcome [12]. By doing so, the algorithm excludes conditions that are no essential part of a sufficient configuration for the respective outcome. As a result, the analysis produces two distinct solutions: the parsimonious solution and the intermediate solution. The parsimonious solution includes all simplifying assumptions derived from counterfactuals in contrast to the intermediate solution which only includes simplifying assumptions based on easy counterfactuals. Hence, the intermediate solution always represents a subset of the parsimonious solution which passed a more thorough reduction procedure. In other words, the data thus provides strong empirical evidence for the causality of these conditions present in the parsimonious solution. This solution thus displays the causal core of a configuration, while the causal periphery includes all conditions present in the intermediate solution [12].

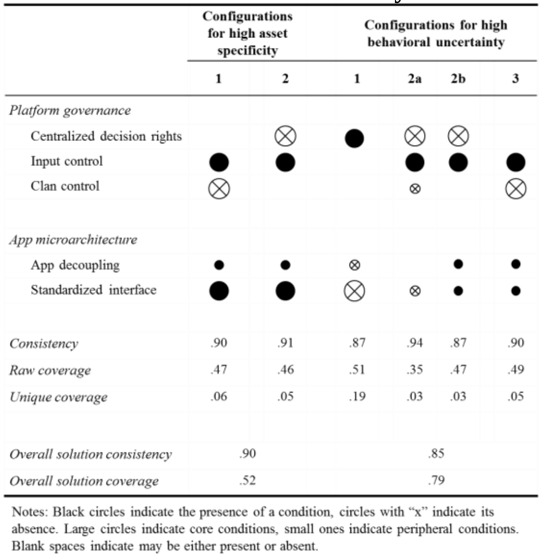
4. Results

The FsQCA approach yielded a number of solutions with considerably high consistency that explains a substantive proportion of complementors' potential hazards. Tables 2 and 3 present the results for high asset specificity, behavioral uncertainty, and technological uncertainty and market uncertainty whereas Tables 4 and 5 display configurations which lead to the non-existence of those hazards. We followed the notation for solution tables by Ragin [28] with black circles ("•") displaying the presence of a condition, crossed-out circles ("⊗") indicating its absence and blank spaces indicate that a condition may be either present or absent. Large Circles thereby refer to core conditions, small circles indicate peripheral conditions. The solutions are grouped on the basis of their core conditions.

4.1. Configurations for high levels of hazard

There are two different core configurations resulting in high asset specificity, three resulting in behavioral uncertainty, four resulting in technological uncertainty and three resulting in market uncertainty (some core configurations split-up into several permutations, these are marked by lower case letters). Consistency for the single configurations ranges from .86 to .94, acceptable levels [28]. The overall consistency values for the four contextual hazards are .90, .85, .85, and .87. Roughly spoken, overall these values display the degree to which the configurations consistently result in the outcome of interest. Therefore, we can say that these five solutions consistently result in high levels of hazard with 90, 85, 85 and 87 percent. Overall solution coverage represents the extent to which the configurations cover cases of the respective outcome [28]. Hence, it explains what percentage of membership for the outcome set can be captured by the configurations of conditions. Thus, cases with high levels of asset specificity, behavioral uncertainty, technological uncertainty and market uncertainty are covered to 52, 79, 82 and 82 percent respectively by the solutions presented in the tables.

Table 2. Configurations for high asset specificity and behavioral uncertainty



	Configurations for high asset specificity		Configurations for high behavioral uncertainty			
	1	**2**	**1**	**2a**	**2b**	**3**
Platform governance						
Centralized decision rights		⨂	●	⨂	⨂	
Input control	●	●		●	●	●
Clan control	⨂			⊗		⨂
App microarchitecture						
App decoupling	•	•	⊗		•	•
Standardized interface	●	●	⨂	⊗	•	•
Consistency	.90	.91	.87	.94	.87	.90
Raw coverage	.47	.46	.51	.35	.47	.49
Unique coverage	.06	.05	.19	.03	.03	.05
Overall solution consistency	.90		.85			
Overall solution coverage	.52		.79			

Notes: Black circles indicate the presence of a condition, circles with "x" indicate its absence. Large circles indicate core conditions, small ones indicate peripheral conditions. Blank spaces indicate may be either present or absent.

By comparing the configurations across all four dimensions of hazards, we found four strong patterns:

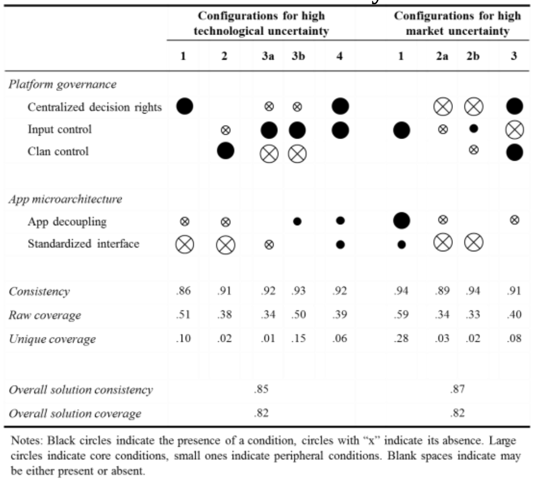
Pattern I) If platform owners heavily rely on mechanisms of input control, this causes high levels of asset specificity, behavioral uncertainty and technological uncertainty from a complementor's viewpoint. In contrast, concerning market uncertainty both its presence (solution 1) and absence (solution 3) are core elements of configurations.

Pattern II) The only configuration which consistently lead to asset specificity, behavioral uncertainty and technological uncertainty involves the presence of input control, the absence of clan control mechanisms and the presence of app decoupling. Hence, under such forms of platform control, the complementor may be unable to lower transaction costs based on own design choices in the microarchitecture.

Pattern III) With exception to solutions 2a and 2b for market uncertainty, hazards from the complementor's perspective are always associated with the platform owner ensuring itself a portion of power within the governance form. This may be either via input control or via centralized decision rights.

Pattern IV) Third-party developers' choice to standardize and formalize the interface of their apps towards a single platform may under certain circumstances be a necessary mechanism to reduce the uncertainty they face when interacting with the platform. However, this comes at the cost of specific investments in their relationship with the platform, so that there may be a trade-off between reducing uncertainty via standardization and vice versa increasing asset specificity by doing so.

Table 3. Configurations for high technological and market uncertainty



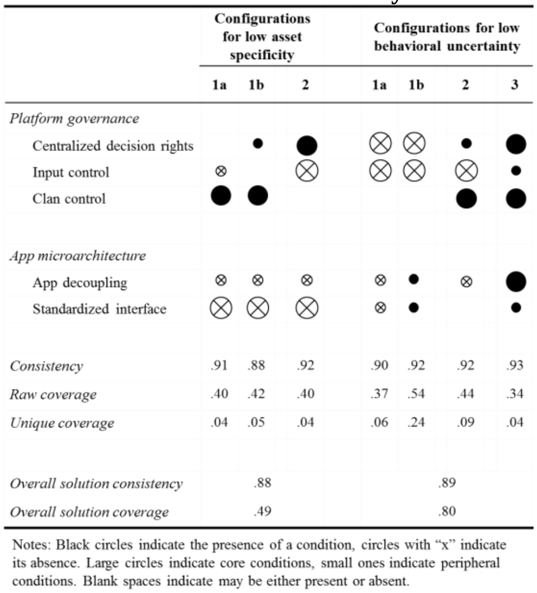
	Configurations for high technological uncertainty					Configurations for high market uncertainty			
	1	**2**	**3a**	**3b**	**4**	**1**	**2a**	**2b**	**3**
Platform governance									
Centralized decision rights	●		⊗	⊗	●		⨂	⨂	●
Input control		⊗	●	●	●	●	⊗	•	⨂
Clan control		●	⨂	⨂				⊗	●
App microarchitecture									
App decoupling	⊗	⊗		•	•	●	⊗		⊗
Standardized interface	⨂	⨂	⊗		•	•	⨂	⨂	
Consistency	.86	.91	.92	.93	.92	.94	.89	.94	.91
Raw coverage	.51	.38	.34	.50	.39	.59	.34	.33	.40
Unique coverage	.10	.02	.01	.15	.06	.28	.03	.02	.08
Overall solution consistency	.85					.87			
Overall solution coverage	.82					.82			

Notes: Black circles indicate the presence of a condition, circles with "x" indicate its absence. Large circles indicate core conditions, small ones indicate peripheral conditions. Blank spaces indicate may be either present or absent.

4.2. Configurations for low levels of hazard

There are two different core configurations resulting in low levels of asset specificity, three resulting in low levels of behavioral uncertainty, four resulting in low levels of technological uncertainty and one resulting in low level market uncertainty. Consistency for the single configurations ranges from .88 to .93. The overall consistency values for the four contextual hazards are .88, .89, .90, and .93. Overall solution coverage for low levels of asset specificity, behavioral uncertainty, technological uncertainty and market uncertainty is at .49, .80, .84 and .59.

Table 4. Configurations for low asset specificity and behavioral uncertainty



	Configurations for low asset specificity			Configurations for low behavioral uncertainty			
	1a	1b	2	1a	1b	2	3
Platform governance							
Centralized decision rights		•	●	⨂	⨂	•	●
Input control	⊗		⨂	⨂	⨂	⨂	•
Clan control	●	●				●	●
App microarchitecture							
App decoupling	⊗	⊗	⊗	⊗	•	⊗	●
Standardized interface	⨂	⨂	⨂	⊗	•		•
Consistency	.91	.88	.92	.90	.92	.92	.93
Raw coverage	.40	.42	.40	.37	.54	.44	.34
Unique coverage	.04	.05	.04	.06	.24	.09	.04
Overall solution consistency	.88			.89			
Overall solution coverage	.49			.80			

Notes: Black circles indicate the presence of a condition, circles with "x" indicate its absence. Large circles indicate core conditions, small ones indicate peripheral conditions. Blank spaces indicate may be either present or absent.

By comparing the configurations across all four dimensions of hazards, we found four strong patterns:

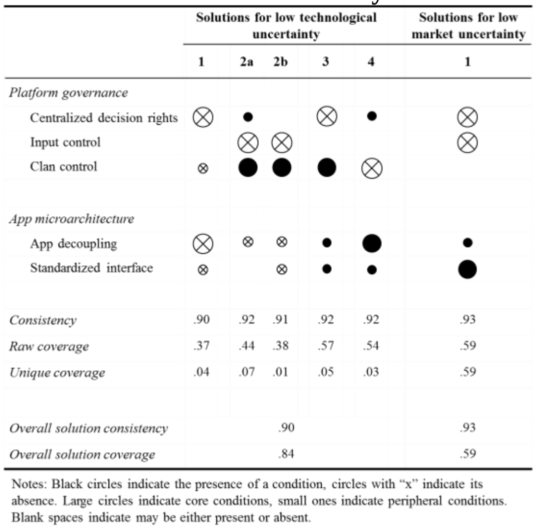
Pattern V) Clan control seems to be an effective mechanism to reduce complementors' transaction costs. Its attenuating effect appears if a) either input control is low or decision making is delegated (for behavioral and technological uncertainty), b) if decision rights are centralized and input is controlled but concurrently app decoupling and interface standardization are present (behavioral uncertainty), and c) if the complementor resigns to standardize its platform-app interfaces (asset specificity).

Pattern VI) If clan control is absent, low levels of hazard are still possible, but only if either the complementor modularizes its app architecture via decoupling or the platform owner does not draw on governance mechanisms which grant it power, namely centralized decision making and input control.

Pattern VII) The absence of both centralized decision rights and input control reduces behavioral and market uncertainty from the complementor's viewpoint.

Pattern VIII) Non-modularization of the app microarchitecture seems to be necessary for low levels of asset specificity whereas low perceived market uncertainty is only achieved consistently if modularization in the form of app decoupling and standardized interfaces is present.

Table 5. Configurations for low technological and market uncertainty



	Solutions for low technological uncertainty					Solutions for low market uncertainty
	1	2a	2b	3	4	1
Platform governance						
Centralized decision rights	⨂	•		⨂	•	⨂
Input control		⨂	⨂			⨂
Clan control	⊗	●	●	●	⨂	
App microarchitecture						
App decoupling	⨂	⊗	⊗	•	●	•
Standardized interface	⊗		⊗	•	•	●
Consistency	.90	.92	.91	.92	.92	.93
Raw coverage	.37	.44	.38	.57	.54	.59
Unique coverage	.04	.07	.01	.05	.03	.59
Overall solution consistency	.90					.93
Overall solution coverage	.84					.59

Notes: Black circles indicate the presence of a condition, circles with "x" indicate its absence. Large circles indicate core conditions, small ones indicate peripheral conditions. Blank spaces indicate may be either present or absent.

5. Insights and implications

Our primary objective in this paper is to analyze the antecedents of complementors' transaction costs in platform ecosystems. In particular, we attempt to shed light on how the interplay between the architecture of an extension and the different governance mechanisms of platform owners facilitate or diminish the emergence of hazards in the relational context which are likely to raise third-party developers' transaction costs.

The results of our configurational analysis provide several interesting insights for both theory and practice. First, while platform owners might benefit from governance mechanisms which grant them power and authority over the development of the ecosystem, these mechanisms are quite likely to confront the platforms complementors with several hazards. Compared to the centralization of decision rights, input control seems to be a particularly important mechanism. If input control is applied, all types of hazards are likely to be high. However, if input control is absent, all types of hazards are likely to be low. The screening and admission procedures of the platform owner consequently require specific investments to meet such criteria. Furthermore, such control fosters uncertainty as it gives the platform owner a certain amount of power. Input control therefore represents an essential parameter which platform owners should calibrate carefully in order to balance own as well as complementors’ costs and thus ensure healthiness and robustness of the ecosystem.

Second, the results indicate that clan control may be an effective mechanism for the platform owner to lower complementors’ hazards. Such norms, mutual values and goals as soft power instrument help to lower asset specific, behavioral and technological uncertainty. If the platform owner abstains from applying just one of those two power ensuring mechanisms, then clan control is a sufficient way to keep hazards and consequently complementor´s transaction costs low. Hence, clan control might to a certain degree be a suitable measure to lower the negative effects of input control and decision rights centralization.

Finally, the complementor itself can influence the emergence of hazards within the transaction atmosphere. However, the impact of modularizing the design of own apps is limited and comes at costs. On the one hand, app decoupling and standardized interfaces may help to reduce behavioral, technological and market uncertainty. This can be explained as app modularization reduces the effects of technological changes within the platform on a single app and ensures that the complementor can counteract to opportunistic behavior and market changes quickly. Still, these effects only apply constantly if input control is not present or if clan control is applied. Hence, design choices by the complementor possess rather low ability to overcome the effects of design choices made by the platform owner. On the other hand, our data provides evidence for app decoupling and interface standardization to be both necessary and sufficient conditions for high levels of asset specificity. This finding seems counterintuitive and calls for further investigations.

In all, we extend theory on platform ecosystems in three ways. First, this study contributes to previous work on factors that influence ecosystem dynamics [8] by extending current perspectives on costs. Therefore, we propose TCT, which IS research traditionally uses in research on outsourcing [2, 41], as a valuable theoretical lens for platform management. Second, we investigate the antecedents of such costs by examining the influence of the interplay between modular systems and different modes of platform governance on four types of hazards which may arise in the relation between complementor and platform. Third and finally, we provide a fresh theoretical and methodological perspective on this topic by acknowledging the configurational nature of platforms and apply FsQCA to detect meaningful combinations of the input variables. We hope that these configurations may serve as holistic templates which will stimulate academic discussion and prove useful for executives in software development and platform managers.

6. References

[1] Armstrong, J.S. and T.S. Overton, "Estimating Nonresponse Bias in Mail Surveys", Journal of Marketing Research, 1977, pp. 396–402.

[2] Aubert, B.A., S. Rivard, and M. Patry, "A Transaction Cost Model of IT Outsourcing", Information & Management, 41(7), 2004, pp. 921–932.

[3] Baldwin, C.Y., "Where Do Transactions Come From? Modularity, Transactions, and the Boundaries of Firms", Industrial and Corporate Change, 17(1), 2008, pp. 155–195.

[4] Benlian, A., D. Hilkert, and T. Hess, "How Open is this Platform? The Meaning and Measurement of Platform Openness from the Complementors’ Perspective", Journal of Information Technology, 30(3), 2015, pp. 209–228.

[5] Boudreau, K.J., "Let a Thousand Flowers Bloom? An Early Look at Large Numbers of Software App Developers and Patterns of Innovation", Organization Science, 23(5), 2012, pp. 1409–1427.

[6] Bresnahan, T. and S. Greenstein, "Mobile Computing: The Next Platform Rivalry", American Economic Review, 104(5), 2014, pp. 475–480.

[7] Casciaro, T. and M.J. Piskorski, "Power Imbalance, Mutual Dependence, and Constraint Absorption: A Closer Look at Resource Dependence Theory", Administrative Science Quarterly, 50(2), 2005, pp. 167–199.

[8] Ceccagnoli, M., C. Forman, P. Huang, and D.J. Wu, "Cocreation of Value in a Platform Ecosystem: The Case of Enterprise Software", MIS Quarterly, 36(1), 2012, p. 263–290.

[9] Cennamo, C. and J. Santalo, "Platform Competition: Strategic Trade-offs in Platform Markets", Strategic Management Journal, 34(11), 2013, pp. 1331–1350.

[10] El Sawy, O.A., A. Malhotra, Y. Park, and P.A. Pavlou, "Research Commentary: Seeking the Configurations of Digital Ecodynamics: It Takes Three to Tango", Information Systems Research, 21(4), 2010, pp. 835–848.

[11] Fama, E.F. and M.C. Jensen, "Separation of Ownership and Control", The Journal of Law & Economics, 26(2), 1983, pp. 301–325.

[12] Fiss, P.C., "Building Better Causal Theories: A Fuzzy Set Approach to Typologies in Organization Research", Academy of Management Journal, 54(2), 2011, pp. 393–420.

[13] Fornell, C. and D.F. Larcker, "Evaluating Structural Equation Models with Unobservable Variables and Measurement Error", Journal of Marketing Research, 1981, pp. 39–50.

[14] Gatignon, H. and E. Anderson, "The Multinational Corporation's Degree of Control over Foreign Subsidiaries: An Empirical Test of a Transaction Cost Explanation", Journal of Law, Economics, & Organization, 4(2), 1988, pp. 305–336.

[15] Ghazawneh, A. and O. Henfridsson, "Balancing Platform Control and External Contribution in Third-party Development: The Boundary Resources Model", Information Systems Journal, 23(2), 2013, pp. 173–192.

[16] Heide, J.B. and G. John, "Alliances in Industrial Purchasing: The Determinants of Joint Action in Buyer-Supplier Relationships", Journal of Marketing Research, 1990, pp. 24–36.

[17] Kirsch, L.J., V. Sambamurthy, D.-G. Ko, and R.L. Purvis, "Controlling Information Systems Development Projects: The View from the Client", Management Science, 48(4), 2002, pp. 484–498.

[18] Kude, T. and J. Dibbern, "Tight Versus Loose Organizational Coupling within Inter-firm Networks in the Enterprise Software Industry – The Perspective of Complementors", AMCIS Proceedings, 2009, Paper 666.

[19] Kude, T., J. Dibbern, and A. Heinzl, "Why Do Complementors Participate? An Analysis of Partnership Networks in the Enterprise Software Industry", IEEE Transactions on Engineering Management, 59(2), 2012, pp. 250–265.

[20] Kumar, N., L.W. Stern, and J.C. Anderson, "Conducting Interorganizational Research Using Key Informants", Academy of Management Journal, 36(6), 1993, pp. 1633–1651.

[21] Leischnig, A. and K. Kasper-Brauer, "Employee Adaptive Behavior in Service Enactments", Journal of Business Research, 68(2), 2015, pp. 273–280.

[22] Lewis, B.R., G.F. Templeton, and T.A. Byrd, "A Methodology for Construct Development in MIS Research", European Journal of Information Systems, 14(4), 2005, pp. 388–400.

[23] Liu, Y., J. Mezei, V. Kostakos, and H. Li, "Applying Configurational Analysis to IS Behavioural Research: A Methodological Alternative for Modelling Combinatorial Complexities", Information Systems Journal, 2015, online version of record published before inclusion in an issue.

[24] Mowery, D.C., J.E. Oxley, and B.S. Silverman, "Strategic Alliances and Interfirm Knowledge Transfer", Strategic Management Journal, 17(S2), 1996, pp. 77–91.

[25] Park, Y. and O.A. El Sawy, “Discovering the Multifaceted Roles of Information Technologies with a Holistic Configurational Theory Approach”, HICSS Proceedings, 2012.

[26] Pierce, L., "Big Losses in Ecosystem Niches: How Core Firm Decisions Drive Complementary Product Shakeouts", Strategic Management Journal, 30(3), 2009, pp. 323–347.

[27] Podsakoff, P.M., S.B. MacKenzie, J.-Y. Lee, and N.P. Podsakoff, "Common Method Biases in Behavioral Research: A Critical Review of the Literature and Recommended Remedies", Journal of Applied Psychology, 88(5), 2003, p. 879.

[28] Ragin, C.C., “Redesigning Social Inquiry: Fuzzy Sets and Beyond”, University of Chicago Press, Chicago, 2008.

[29] Rindfleisch, A. and J.B. Heide, "Transaction Cost Analysis: Past, Present, and Future Applications", Journal of Marketing, 1997, pp. 30–54.

[30] Rochet, J. and J. Tirole, "Platform Competition in Two-sided Markets", Journal of the European Economic Association, 1(4), 2003, pp. 990–1029.

[31] Sanchez, R. and J.T. Mahoney, "Modularity, Flexibility, and Knowledge Management in Product and Organization Design", Strategic Management Journal, 17(S2), 1996, pp. 63–76.

[32] Schneider, C.Q. and C. Wagemann “Set-theoretic Methods for the Social Sciences: A Guide to Qualitative Comparative Analysis”, Cambridge University Press, 2012.

[33] Stump, R.L. and J.B. Heide, "Controlling Supplier Opportunism in Industrial Relationships", Journal of Marketing Research, 1996, pp. 431–441.

[34] Sutcliffe, K.M. and A. Zaheer, "Uncertainty in the Transaction Environment: An Empirical Test", Strategic Management Journal, 19(1), 1998, pp. 1–23.

[35] Tiwana, A., "Evolutionary Competition in Platform Ecosystems", Information Systems Research, 26(2), 2015, pp. 266–281.

[36] Tiwana, A., "Platform Desertion by App Developers", Journal of Management Information Systems, 32(4), 2015, pp. 40–77.

[37] Tiwana, A., B. Konsynski, and A.A. Bush, "Research Commentary: Platform Evolution: Coevolution of Platform Architecture, Governance, and Environmental Dynamics", Information Systems Research, 21(4), 2010, pp. 675–687.

[38] Walker, G. and D. Weber, "A Transaction Cost Approach to Make-or-buy Decisions", Administrative Science Quarterly, 1984, pp. 373–391.

[39] Wareham, J., P.B. Fox, and J.L. Cano Giner, "Technology Ecosystem Governance", Organization Science, 25(4), 2014, pp. 1195–1215.

[40] Wathne, K.H. and J.B. Heide, "Opportunism in Interfirm Relationships: Forms, Outcomes, and Solutions", Journal of Marketing, 64(4), 2000, pp. 36–51.

[41] Watjatrakul, B., "Determinants of IS Sourcing Decisions: A Comparative Study of Transaction Cost Theory versus the Resource-based View", Journal of Strategic Information Systems, 14(4), 2005, pp. 389–415.

[42] Williamson, O.E., “The Economic Intstitutions of Capitalism”, Simon and Schuster, 1985.

[43] Williamson, O.E., "Comparative Economic Organization: The Analysis of Discrete Structural Alternatives", Administrative Science Quarterly, 1991, pp. 269–296.

[44] Williamson, P.J. and A. de Meyer, "Ecosystem Advantage", California Management Review, 55(1), 2012, pp. 24–46.

[45] Yoo, Y., R.J. Boland Jr, K. Lyytinen, and A. Majchrzak, "Organizing for Innovation in the Digitized World", Organization Science, 23(5), 2012, pp. 1398–1408.

[46] Zhang, Y., W. Zhong, N. Wen, and D. Jiang, "Asset Specificity and Complementarity and MNE Ownership Strategies: The Role of Institutional Distances", Industrial Marketing Management, 43(5), 2014, pp. 777–785.