

Please quote as: Dellermann, D. & Reck, F. (2017): Minimizing Complementors' Risk in Third-Party Innovation: A Qualitative Comparative Analysis (QCA) of Digital Platform Configurations. In: Hawaii International Conference on System Sciences (HICSS). Waikoloa, HI, USA.

Minimizing Complementors' Risk in Third-Party Innovation: A Qualitative Comparative Analysis (QCA) of Digital Platform Configurations

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

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

Abstract

The emergence of platforms is shifting the locus of digital innovation to ecosystems on which numerous developers create extensions with additional functionalities. Despite all the potential benefits for complementors, however, this new organizing logic of digital innovation also introduced essential new risks. Recent studies in IS focused on risk of IT projects from a contingency perspective neglecting the complexity of ecosystems. In order to shed light on this, our work examines how app architecture as a complementor's control mechanism and four types of ecosystem hazards shape the likelihood and impact of the risk of failure in third-party innovation. 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

Pervasive digital technology significantly changes the logic of innovation. One of the most important aspects of organizing such innovation processes is shifting the locus of innovation on technological platforms [46, 59]. A digital platform, i.e. an extensible code base, allows the development of complementary products or services (e.g. applications) that augment a platform's native functionality [50]. Companies offering such complementary applications are called complementors or third-party developers [18]. To accelerate innovation on digital platforms, platform owners have to create and sustain vibrant ecosystems of third-party developers [9]. Modular platform architecture enables complementors to develop their own apps independently, yet platform interfaces

ensure their interoperability. This tendency towards a disintegrated architecture is mirrored by an increasing degree of interorganizational modularity, distributing the partitioning of innovation among many heterogeneous firms [5].

Digital technology therefore creates several idiosyncrasies in the organizational logic of innovation [50]. First, the loosely coupled relationships between actors like the platform owner and single third-party developers represent a hybrid form of organizations which exhibits characteristics of both markets and formal alliances in the traditional sense of economic exchange theories [49]. Second, following this logic, control and knowledge is distributed between various actors [50]. Finally, such relations are frequently characterized by coopetition (i.e. simultaneous cooperation and competition). For instance, although platform owners encourage the development of third-party innovations, they might compete with complementors in certain market niches [12].

Although organizing digital innovation around a technological platform has created new business opportunities by providing complementary resources, it also introduced essential new risks. We refer to this phenomenon as risk of third-party innovation. In comparison to traditional risks of software engineering [7, 48], the locus of this form of risk is not within the own organizational boundaries but on platforms as well as within the focal complementor's relationship multiple and heterogeneous actors. Exogenous and relation-specific factors like for instance opportunistic behavior of the platform owner, market related factors as well technological dependencies on the platform, thus constitute crucial threats which lay outside the direct control of a complementor.

In order to theoretically explain the emergence of software development risks and provide IS management with means for its management, previous research proposes that successful organizations establish a fit between the degree of uncertainty of their environment and their structural

and control approaches [10]. This perspective extensively examined the role and interplay of control mechanism and environmental factors in influencing the risk of IT projects [34, 37].

In the context of third-party development on technological platforms, this perspective runs its limits for two main reasons. First, the contingency approach assuming the existence of a single state of fitness between control mechanisms and potential exogenous hazards is not able to capture the increasing dynamics and complexity of an ecosystem as the focus of IT innovation is shifting to platforms. We therefore utilize configuration theory [33] as theoretical lens to overcome the traditional reductionism problem [27] and examine the equifinality of different solutions for managing risk in ecosystems where a different set of elements can produce the same outcome.

Second, complementors are typically not able to apply direct control mechanisms to govern third-party innovation in platform ecosystems for reducing their risk. Congruent with previous work, which highlights the role of modular architecture as a control function for alliances [43] or to reduce opportunistic behavior [20] we argue that the modularization of application-platform linkages is the useful mechanism for complementors to manage the relation with the platform owner.

Addressing these two shortcomings of previous research, the purpose of our work is therefore to shed light on complementors' third-party innovation risk by explaining its prevalence based on different configurations exogenous hazards from the platform ecosystem as well as the microarchitecture of single applications which may serve as a safeguard against those hazards. We therefore address the following research question: *Which configurations of architectural choices and ecosystem-related hazards minimize the complementor's risk of third-party innovation?*

To answer these questions, our research analyzes data from a survey of 42 complementors on five leading cloud platforms using fuzzy set Qualitative Comparative Analysis (FsQCA) [33]. The FsQCA approach is a case-oriented method that enables analyzing asymmetric and complex causal effects by extracting configurations that consistently lead to the outcome of interest [14, 13].

Our study offers three noteworthy contributions. First, it outlines the influence of environmental hazards on the risk related to a major form of organizing digital innovation, platform-based application development. Second, it empirically validates the inseparability of environmental dynamics and architectural choices in such digital

innovation settings. Third, it offers insights on how digital architecture can be utilized as a coordination device of complementors to manage interorganizational relations and to reduce risk.

2. Conceptual background

2.1. Risk of organizing third-party innovation

In IS research, risk represents a function of both uncertainty and some kind of loss or damage, which is experienced by a decision maker [26]. A further crucial concept in this context is hazards, which is defined as a source of danger [21]. Consequently, if an actor is not able safeguard against such hazards, they create a potential loss, i.e. risks.

Previous approaches examining risk in inter-organizational arrangements like for instance R&D alliances [e.g. 31] or IT outsourcing [e.g. 3] are theoretically grounded in theories of economic exchange (i.e. transaction cost theory [49]). Following the logic stated in the introduction, however, we argue that the specific characteristics of digital technologies create also significant changes in the nature and analysis of risk. The loosely coupled relationships between the platform owner and a complementor represent a hybrid between characteristics of a market and an alliance. Therefore, significantly new uncertainties evolve for the participants of platform ecosystems. In particular, the distribution of control and knowledge among heterogeneous actors accelerates uncertainty regarding the technology itself or the behavior of the alter [12, 50]. For instance, the platform owner's control over boundary resources (i.e. software development kit (SDK) application programming interfaces (APIs)) makes complementors increasingly dependent [18]. This limits third-party developers' space to control the exchange with the platform owner itself. Furthermore, as this new organizing logic of digital innovation frequently requires cooperation (i.e. simultaneous cooperation and competition) to drive innovation, complementors may suffer from platform owners to adopt and modify their applications in order to capture attractive market niches [12]. While platform owners encourage the development of third-party innovations, the loss of intellectual property is therefore a common threat in this context [6].

The risk of third-party innovation as an outcome variable is therefore defined as the potential failure of the complementor's innovation effort in a loosely coupled and cooperative relationship with the platform owner. This concept has two distinctive sub dimensions [31]: risk likelihood (i.e. the probability that the digital innovation effort will fail) and risk

impact (i.e. the perceived possible loss in the form of missing or underachieving the goals of the innovation effort). While the first sub dimension is mainly resulting from uncertainty, the latter is accelerated by the specificity of a digital platform and the resulting migration costs to another technology.

2.2. A configurational perspective on organizing digital innovation

In IS, researchers adopt a contingency approach risk management to examine the role and interplay of control mechanism and environmental factors in influencing the risk of IT projects [35, 38]. This approach has been strongly influenced by research in organizational contingency theory, which proposes that successful organizations ensure a unifinality of fit between the degree of uncertainty of their environment and their structures [11]. Rather than assuming the existence of best-fitting combinations of predictor variables, we assume equifinality of different configuration of variables. Thereby, we take a holistic viewpoint which abstains from evaluating net effects of single variables but treats such configurations in a whole as explanatory factors for the outcome of interest. Such an application of configurational theory in the context of digital innovation in platform ecosystems is suitable for two reasons.

First, in configurational approaches whole sets of elements serve to simultaneously explain the outcomes of interest [13]. Because of that, configurational theory is particularly appropriate to explain synergetic and complementary causalities [33]. This resonates well with current theoretical perspectives on the organizing logic of digital innovation in general and platform and ecosystem management in specific. Research in this field highlights the inseparability of ecosystem dynamics from app architectures and their mutual effect on innovation outcomes. Therefore, examining variable in isolation therefore is no reasonable approach towards explaining risk in third-party development.

On the other hand, recent organizational [14] and information systems research [25] suggests that the assumption of symmetric causal relationships might not adequately display organizational realities [13]. In contrast, configurational theories imply equifinality between different sets of initial conditions [33] and assume asymmetric rather than symmetric relations between conditional variables and outcomes [13]. Consequently, corresponding analysis procedures allow for the detection of sufficient or necessary causes of a dependent variable. For instance, while the existence of a

particular hazard might consistently lead to high risk for complementors, this does not mean that its absence will lead to low levels of risk (e.g., there might be other hazards which substitute for it). Considering these advantages of configurational perspective, we argue that understanding organizational outcomes of the distributed organizing logic of digital innovation strongly depends on configuration of several design choices with its environment.

2.3. Research framework

Fig. 1 illustrates the framework of this article. We divided the concept of third-party innovation risk into two distinctive dimensions: risk likelihood (i.e. the probability that the digital innovation effort will fail) and risk impact (i.e. the perceived possible loss). The framework comprises two facets of causal conditions for risk. It proposes that from the perspective of complementors, the configuration of four exogenous hazards (i.e. platform specificity; behavioral, market & technological uncertainty) and two endogenous choices to manage their innovation effort (i.e. app decoupling and standardization of interfaces) influence the risk of third-party innovation.

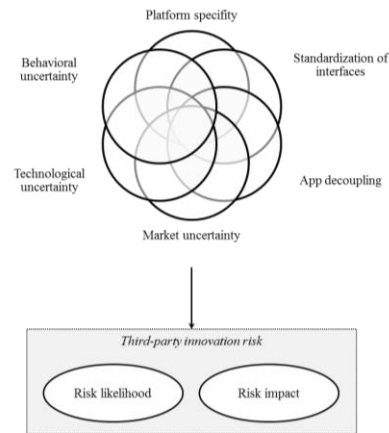


Figure 1. Research framework

In the selection of our causal conditions, we follow notions of Tiwana et al. [46] on intra-platform dynamics and the required fit of architecture and environmental dynamics to process strategic outcomes. Our set of causal conditions therefore includes design elements outside (hazards of the ecosystem) as well as within (app decoupling and standardized interfaces) the range of complementors' influence and is theoretically guided by the dimensions of transaction cost theory [49].

Platform specificity: The specificity of a certain platform represents the first hazard for a

complementor. Platform specificity refers to the transferability of a complementor's application to a different platform [4] as well as the value of complementor's assets within alternative partner relations [36]. For instance, platforms require investments in relation-specific knowledge to participate in the platform ecosystem and capitalize from the access to complementary resources and capabilities [3]. Specific assets can be for instance, human assets, technological assets or knowledge about platform architecture, interface specifications and market characteristics. High levels of asset specificity and the related investment requirements create dependence between partners, lead to lock-in effects which make it difficult for the complementors and move to another platform [22]. A high specificity of assets required for building complementary products therefore results for instance in high multi-homing costs [2]. Therefore the amount of a potential loss is likely to be higher under conditions of high platform specificity.

The second exogenous hazard for complementors in platform ecosystems is uncertainty, which is most commonly defined as the absence of complete information about the contextual environment. This in turn leads to an inability to predict it accurately [29]. The concept of uncertainty is crucial in organization theory and frequently applied in studies on risk in IS [29]. For the purpose of our study we define uncertainty rather on the interorganizational environment than on the project level. On this level we apply an environmental perspective on uncertainty, which explains the unpredictability of the firm's environment surrounding a relationship between firms [16, 44].

Market uncertainty: Market conditions are crucial drivers for the risk of complementors, as for instance the sustainability of the specific niche is required to succeed. Volatile customer demand, the unpredictable emergence of new substitute products or changes in the competitive environment might increase the threat of failure during the development of complementary products.

Technological uncertainty: Furthermore, technological unpredictability covers the inability to accurately forecast the technological requirements within the relationship, which is especially important in complementary platform markets. Technological complexity and changes are the most significant sources of uncertainty [29]. Technological uncertainty is also frequently related to a lack of experience with the technologies employed in the ecosystem [30], which increases the threat of failure due to inadequate capabilities. Furthermore, the unpredictability of technological evolution might

constitute a source of risk during third-party innovation [46].

Behavioral uncertainty: In contrast to environmental uncertainty, which is not directly related to the partner, behavioural uncertainty arises from the complexity and difficulty of evaluating each other's actions within a relationship. Taken to the platform context, the platform owner might follow its individual interests and cause hidden costs by inefficient and ineffective behavior [49]. Moreover, although platform owners encourage the development of complementary products to nurture the overall value of the ecosystem [37], there is often a tension between them and complementors. This tension arises from the complementor's threat of opportunistic behavior of the platform owner by for instance exploiting resources or competing in the partner's niche [22].

Building on Tiwana [46], who outlines the required fit of application architecture and platform dynamics we extend this line reasoning to the risk of third-party innovation. Prior works highlight that the role of modular architecture as control mechanism to influence the outcome of interorganizational arrangements [43] or to reduce opportunistic behavior [20]. Therefore, third-party developers possess design alternatives based on which they can influence the governance of 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 risk by exploiting the benefits of modularization [44, 45]. On the micro level of application architecture, we focus on the modularization of the app-platform linkages rather than internal modular app architectures. App modularization therefore minimizes the application-platform dependencies on the degree to which an app is required to be conforming to the specified interface that is vice versa determined by the platform owner [44, 45]. Hence, applications within the same ecosystem can significantly vary in their level of modularization [28] as its micro-architecture reflects an endogenous choice of the complementor.

App Decoupling: Decoupling allows for changes within a module which do not require parallel changes in the platform and vice versa. App decoupling reduces dependencies at the boundary between app and platform and minimizes the interactions between both [46]. Hence, the technological volatility of a platform does not necessarily require changes in the single application. It enables the flexible and independent development of apps. Third-party developers are therefore able to adapt the application's internal implementation

without the need of knowledge about internal details of the platform [39].

Standardized interfaces: Standardization refers to the use of standards and protocols predefined by the platform owner (e.g., platform specific APIs) that are applied to meet conformance between the platform and the complementor’s applications. Such standards are introduced by a platform owner to manage the relationships between the app and the platform. Standardization reduces the need for iteration between the complementor and the platform owner and ensures interoperability between the platform and the app. This underlines the role of standardized interfaces as a control mechanism [44, 45].

Both mechanisms allow complementors to develop apps independently and ensure interoperability with the platform and represent an architectural control mechanism to manage their innovation activities in the ecosystem.

3. Research methodology

3.1. Data collection and sample description

Our sample of firms 750 firms which are members of five leading cloud platforms (i.e. Microsoft Azure, Oracle Cloud Platform, Amazon Web Services, SAP HANA, and Salesforce Force.com). There were two reasons for choosing these particular platforms. First, all platforms are well-established and have solid traction among third-party developers. Second, in all five platforms, a high level of power imbalance is prevalent, so that they perfectly meet our requirements for analyzing asymmetric third-party relationships.

Key informant data was collected via a web crawling approach which randomly gathered contacts from the platforms’ app stores. This approach is consistent with previous surveys of third-party developers [8]. The potential respondents were contacted via an e-mail containing information on the research project, a link to the online questionnaire as well as the request to complete the survey or to forward the questionnaire to other executives (C-level; IT executives) as further potential key informants [23].

In total, we obtained complete data on N=42 cases. This equals a response rate of 5.6 %, a common value in such settings [e.g. 8]. We assessed this possibility by comparing responses of early and late respondents [2]. T-tests did not reveal any significant differences ($p > 0.05$) rejecting the presence of non-response bias in our dataset.

Complementors from all five platforms replied (Microsoft Azure: 9; Oracle Cloud Platform: 4; Amazon Web Services: 2; SAP HANA: 9; and Salesforce Force.com: 14). Most of them were high-

level executives (C-level: 71.4 %; BU executives: 19 %) and indicated high experience in managing platform-based software development (>10 years: 83.3 %).

3.2. Measurement validation

Based pilot study with managers in the software industry, we constructed our measurement instrument. In order to ensure validity, reliability as well as rigor of our research [24], we adapted existing scales to the platform context and refined them based on the insights from the pilot study. Subsequently, these refined items were evaluated in a pre-test procedure. This helped us ascertaining that the formulation of all items was unambiguous and comprehensible.

Table 1 displays the psychometric statistics of the measured constructs. There is strong evidence for adequate reliability with Cronbach’s α greater than .85 for all variables. Furthermore, we can assert discriminant validity as confirmatory factor analysis yielded adequately high factor loadings concerning so that the Fornell/Larcker criterion is fulfilled for all our study variables [13].

Table 1. Construct measures

Construct	Loadings Range	Cronbach's Alpha	Composite Reliability	AVE
Outcome Variable				
<i>Risk Likelihood</i> Adapted from [31]	.777-.924	.861	.901	.729
<i>Risk Impact</i> Adapted from [31]	.936-.950	.875	.941	.889
Causal Conditions				
<i>Platform Specificity</i> Adapted from [19]	.692-.904	.861	.901	.648
<i>Behavioral Uncertainty</i> Adapted from [42]	.787-.955	.892	.926	.759
<i>Technological Uncertainty</i> Adapted from [47]	.845-.934	.925	.947	.818
<i>Market Uncertainty</i> Adapted from [42]	.844-.944	.919	.943	.806
<i>App Decoupling</i> Adapted from [45]	.857-.962	.891	.932	.821
<i>Standardization of Interfaces</i> Adapted from [45]	.810-.927	.892	.926	.757

To reject the possibility of common method bias, we conducted Harman’s one-factor test [27]. The unrotated factor solution resulted in 5 factors explaining 77 % of the variance (35 % was the largest variance explained by one factor). Hence, common method bias is unlikely to be a problem.

3.3. Fuzzy-set QCA

We chose FsQCA as means to analyze the obtained data. This set-theoretic approach is utmost suitable to configurational theories as it aims at extracting whole configurations rather than single factors that help to explain outcomes of interest [14]. Thereby, FsQCA draws on set-based measures of

consistency and coverage in order to evaluate the predictive power of the potentially possible conditional configurations. Consistency values display to which degree cases that share a certain combination of conditions also lead to a specific outcome [33]. Hence, this indicator is analogous to correlation estimates in statistical methods. The other indicator of quality, coverage, represents the degree to which a configuration covers the instances on which a specific outcome is realized. Defined as such, the meaning of coverage values resembles that of R-square values in regression analysis. The FsQCA procedure consists of three steps through which consistent configurations are detected [33]: calibration, construction of truth tables, truth table analysis.

Calibration of construct measures is necessary because FsQCA as a set-theoretic analysis approach draws on membership scores (here, e.g. membership in the group of firms with highly decoupled apps) rather than values on interval or ratio scales. In our study, we thus transformed the Likert scale measures into fuzzy set membership scores. These range between 0 and 1 with 0 indicating full non-membership, 1 indicating full membership and 0.5 marking the crossover point [40]. We follow the calibration approach outlined by Fiss [14] and chose the observed maximum and minimum values within the sample to specify full membership and full non-membership for all variables. The median of observed values served as cross-over point. Based on these three values, the calibration procedure in the FsQCA software program (version 2.5) [34] transforms all obtained measures to membership scores.

The second step of FsQCA is the construction and refinement of a matrix of all possible configurations of antecedent conditions (in our case a 64x6 matrix; in general $2^k \times k$, with k as the number of conditions observed [33]). In order to fit the requirements of FsQCA, this truth table must subsequently be refined. This procedure evaluates each possible configuration on the basis of two criteria: frequency and consistency. The frequency assesses which of the possible configurations actually appear in the dataset. In Large samples, it is often reasonable to exclude infrequent cases so that it is necessary to set a frequency threshold for the inclusion of configurations in the further analysis procedure. As our sample is medium-sized in terms of FsQCA literature, we chose the standard threshold of 1 which is suitable for samples of this size [40]. The consistency criterion captures if a truth table row consistently yields an outcome. The consistency value thereby should outreach at least .8 [33], so we

chose a rather conservative threshold of .9. Overall, in 28 cases, configurations exceeded the frequency threshold of which 13 also exceeded the consistency threshold for risk likelihood and 17 for risk impact.

In the third step, the truth tables are analyzed via counterfactual analysis. This approach is based on Boolean algebra in general and applies the Quine-McCluskey algorithm in particular. This algorithm strips away factors which are not consistently present concerning a particular outcome [14] in order to identify the conditions within a configuration which cause the outcome. Hence, the algorithm excludes conditions that are no essential part of a sufficient configuration for the respective outcome and produces two distinct solutions: the parsimonious solution and the intermediate solution. The parsimonious solution on the one hand draws on all simplifying assumptions derived from counterfactuals. It passes a more thorough reduction procedure, so that the data provides strong empirical evidence for the causality of these conditions. Therefore, the parsimonious solution encompasses the causal core of conditional variables. In contrast, the intermediate solution only includes simplifying assumptions based on easy counterfactuals [33]. The conditional variables which appear in the intermediate solution but do not appear in the parsimonious solution thus represent the causal periphery of a configuration [14].

4. Results

The results of the FsQCA reveal several patterns that explain how different configurations of app architecture and environmental hazards result in high or low levels of both risk likelihood and risk impact. We extracted these patterns by comparing structures of different configurations [14]. Figure 2-5 show the configurations resulting from FsQCA. Black circles indicate the presence of a condition, crossed-out circles indicate the absence of a condition, large circles indicate core condition, and small circles indicate peripheral conditions. Blank spaces indicate a condition may be either present or absent.

4.1. Configurations for achieving high likelihood of risk

We identified seven different configurations that result in a high likelihood of risk. Consistency for configurations ranges from 0.90 to 0.99. Raw coverage, which describes the importance of a certain configuration in explaining the intended outcome, range from 0.26 to 0.46. The overall solution consistency shows these seven solutions can consistently result in high likelihood of risk with 89

% . The overall solution coverage indicates that the extent to which these seven configurations cover high likelihood of risk cases is 76 %. We compared the seven configurations of our analysis to extract two strong patterns:

- I) In platform ecosystems with a high level of market uncertainty complementors are very likely to perceive a high likelihood of risk in third-party innovation (2a&b; 3a&b), which can be explained by the increased likelihood for market disruption or instability of the complementor’s niche.
- II) If the interfaces are not standardized and market uncertainty is high (3a&b), especially with lack of app decoupling as peripheral condition, the likelihood of risk for complementors is high as changing market conditions might increase the need for adaptations in the application. However, if apps are not modularized, complementors are not able to improve the application fast and independently. Therefore, lack of modularization reduces the flexibility to react to changes within the market environment.

Hazards	Configurations for high risk likelihood						
	1a	1b	1c	2a	2b	3a	3b
Platform specificity	⊗	⊗	●	⊗	●	●	⊗
Behavioral uncertainty	⊗		●	⊗	●	⊗	●
Technological uncertainty		●	●	⊗	⊗	⊗	⊗
Market uncertainty	⊗	⊗	●	●	●	●	●
<i>App microarchitecture</i>							
App decoupling	⊗	⊗		⊗	●	⊗	⊗
Standardized interface	⊗	⊗	⊗	●	●	⊗	⊗
Consistency	.90	.92	.99	.92	.96	.98	.98
Raw coverage	.40	.39	.46	.32	.34	.29	.36
Unique coverage	.03	.01	.15	.03	.08	.01	.01
Overall solution consistency				.89			
Overall solution coverage				.76			

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.

Figure 2. Configurations for high risk likelihood

4.2. Configurations for achieving high impact of risk

Furthermore, we identified seven different configurations that result in a high impact of risk that exceed minimum consistency threshold. These seven solutions consistently result in high risk impact with 89 % and cover 81 % of cases with this outcome. Comparing the seven configurations reveals two further important patterns:

- III) The impact of complementor’s risk in third-party innovation is high when the environment

is volatile. In particular, market uncertainty (2a, b, c; 4a & b) and technological uncertainty (1; 4a&b) are the main hazards to result in a high impact of risk.

- IV) The interplay of high interface standardization and low app decoupling (3) represents the second pattern to create a high impact of risk. This can be explained as high standardization requires high investment of the complementor to adhere platform-specific interface standards while a lack of decoupling reduces flexibility and increases the threat of cascading ripple effects that might disrupt its interoperability with the platform.

Hazards	Configurations for high risk impact						
	1	2a	2b	2c	3	4a	4b
Platform specificity	⊗	●	⊗	●	⊗		●
Behavioral uncertainty		⊗	●	●	⊗	●	
Technological uncertainty	●	⊗	⊗		⊗	●	●
Market uncertainty	⊗	●	●	●		●	●
<i>App microarchitecture</i>							
App decoupling	⊗	⊗	⊗	●	⊗	●	●
Standardized interface	⊗	⊗	⊗	●	●	⊗	●
Consistency	.94	.96	.95	.99	.91	.92	.98
Raw coverage	.35	.24	.30	.48	.32	.37	.48
Unique coverage	.03	.00	.01	.04	.01	.06	.05
Overall solution consistency					.89		
Overall solution coverage					.81		

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.

Figure 3. Configurations for high risk impact

4.3. Configurations for achieving low likelihood of risk

Figures 4 and 5 show the configurations for a low level of risk. We compared these sets of causal conditions with the configurations that lead to high risk to detect relevant differences. Consequently, we identified six configurations that result in a low likelihood of risk. These solutions consistently result in a low likelihood of risk with 91 % and cover 72 % of cases with this outcome. Comparing the six sets of causal conditions we extracted three further patterns:

- V) If behavioral uncertainty is missing, complementors perceive a low likelihood of risk (1a&b; 2a&b), although technological uncertainty is high (1a&b). This shows that complementors that are able to monitor the behavior of the platform owner face a lower likelihood of risk as they reduce the space for opportunism.

- VI) Configurations of market uncertainty in presence with an absence of technological uncertainty account for low risk likelihood (3a&b) if the company does not draw on app decoupling. This fact can be explained as technological stability allows the complementor to reduce risk by offering ability to react to changes in the market quickly. Under these circumstances app decoupling does not offer additional benefits.
- VII) Likelihood of third-party innovation risk is low when interfaces are highly standardized (2a&b), which reflects the role of interfaces to standardize rules that apps ought to obey and can expect the platform to obey. This underlines the role of app architecture as a control mechanism for risk.

	Configurations for low risk likelihood					
	1a	1b	2a	2b	3a	3b
<i>Hazards</i>						
Platform specificity	⊗	●	⊗		⊗	●
Behavioral uncertainty	⊗	⊗	⊗	⊗	⊗	●
Technological uncertainty	●	●	⊗	⊗	⊗	⊗
Market uncertainty	⊗	●		⊗	●	●
<i>App microarchitecture</i>						
App decoupling	⊗	●	⊗	●	⊗	⊗
Standardized interface	⊗	●	●	●	⊗	⊗
<i>Consistency</i>	.92	.97	.95	.96	.95	.97
<i>Raw coverage</i>	.30	.30	.34	.49	.25	.29
<i>Unique coverage</i>	.02	.05	.05	.19	.01	.02
<i>Overall solution consistency</i>						.91
<i>Overall solution coverage</i>						.72

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.

Figure 4. Configurations for low risk likelihood

4.4. Configurations for low impact of risk

By analyzing cases for a low impact of risk, we uncovered six different configurations that result in a low impact of risk. These solutions consistently result in that outcome with 90 % and cover 83 % of cases with a low level of risk impact. By comparing these configurations for low risk impact, we found two final patterns:

- VIII) Surprisingly, the specificity of a platform is not a main driver of risk impact but its missing predicts low impact of potential losses (1; 3a&b; 4). From this finding we can derive that complementors do not perceive failure to have a high impact on them when they did not heavily invested in knowledge and other

resources that are idiosyncratic for this certain platform or app migration to another platform can be easily achieved.

- IX) If uncertainty in the ecosystem is low, complementors face a low level of risk impact. Especially, when behavioral and technological uncertainty are missing (2; 4; 5). This shows the interplay of a reduced space for opportunism and the stability of the platform in reducing risk.

	Configurations for low risk impact					
	1	2	3a	3b	4	5
<i>Hazards</i>						
Platform specificity	⊗	●	⊗	⊗	⊗	
Behavioral uncertainty	●	⊗			⊗	⊗
Technological uncertainty	⊗	⊗	●	●	⊗	⊗
Market uncertainty	●	●	⊗	⊗		⊗
<i>App microarchitecture</i>						
App decoupling	⊗	⊗	⊗	⊗	⊗	●
Standardized interface	⊗	⊗	⊗	⊗	●	●
<i>Consistency</i>	.98	.97	.97	.98	.89	.93
<i>Raw coverage</i>	.37	.29	.42	.44	.37	.55
<i>Unique coverage</i>	.02	.01	.03	.03	.03	.26
<i>Overall solution consistency</i>						.90
<i>Overall solution coverage</i>						.83

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.

Figure 5. Configurations for low risk impact

4.5. The drivers of risk and the role of app architecture

From the nine pattern identified in the comparison of configurations that lead to high and low risk, we are able to reveal holistic insights of the drivers of third-party innovation risk and the role of app architecture as a control mechanism. Based on the commonalities among the patterns, we identified three holistic findings to explain the risk of third-party innovation and its management.

First, uncertainty of the platform owner's behavior as well as the specificity of a platform, are no main drivers of complementor's risk. Instead configurations in which both are absent display a low impact and likelihood of risk during digital innovation. Hence, while environmental hazards are needed to turn specific assets and opportunistic partners into considerable drivers of risk, engaging with reliable partners or acting on platform with low asset specificity might at least partially mitigate the impact of environmental hazards.

Second, market and technological uncertainty are the main drivers of risk in digital innovation.

Unstable market conditions and technological volatility are crucially influencing the impact and likelihood of risk during third-party innovation.

Third, application architecture represents not a direct control mechanism to govern the platform dependencies during digital innovation. Standardization of interfaces might rather represent a necessary condition to achieve a low level of risk under certain circumstances. Consequently, the use of standardized interfaces is required to minimize risk. However, if apps are highly modularized, this does not necessarily imply low levels of risk but the effect rather depends on the environment.

5. Conclusion

By comparing different configurations that result in high and low risk, we identified nine patterns that describe the role of environmental hazards and app architecture in shaping risk. From these patterns we derive the role of technological and market uncertainty as core drivers of risk. Furthermore, our findings reveal that behavioral uncertainty and platform specificity are not drivers of risk per se. However, their absence is required to achieve low levels of risk. In addition, we detect the role of app architecture as a control mechanism for third-party innovation. As the absence of app modularity is always implying a high level of risk, it is a necessary condition for minimizing risk.

Therefore, the contribution of our study is threefold. First, it contributes to research of risk in IS by applying a configurational perspective on the new organizing logic of digital innovation and providing evidence for the equifinality of different paths in reducing risk. Second, our research contributes to past work on platform dynamics [46, 13] and intra-platform management [44, 45] by uncovering the interplay of environmental factors and technological architecture in achieving organizational outcomes. Third, we contribute to previous studies on modularization as control mechanism [43, 44, 45] by revealing app modularization as necessary condition to minimize risk.

From a practical point of view, our results show that app developers should use app decoupling and standardized interfaces to reduce risk particular in uncertain environments. Further research in this direction could possibly focus on the interplay of app architecture and governance mechanism, which are introduced by the platform owner, to provide an even more holistic approach to the risk of third-party innovation.

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] Armstrong, M. and J. Wright, "Two-sided Markets, Competitive Bottlenecks and Exclusive Contracts", *Economic Theory*, 32(2), 2007, pp. 353–380.
- [3] Aubert, B.A., S. Rivard, and M. Patry, "A Transaction Cost Model of IT Outsourcing", *Information & Management*, 41(7), 2004, pp. 921–932.
- [4] Baldwin, C.Y. and K.B. Clark, "Design rules: The Power of Modularity", MIT Press, Cambridge, 2000.
- [5] Baldwin, C.Y. and K.B. Clark, "The Architecture of Participation: Does Code Architecture Mitigate Free Riding in the Open Source Development Model?", *Management Science*, 52(7), 2006, pp. 1116–1127.
- [6] Baldwin, C.Y. and J. Henkel, "Modularity and Intellectual Property Protection", *Strategic Management Journal*, 36(11), 2015, pp. 1637–1655.
- [7] Barki, H., S. Rivard, and J. Talbot, "Toward an Assessment of Software Development Risk", *Journal of Management Information Systems*, 10(2), 1993, pp. 203–225.
- [8] Benlian, A., D. Hilbert, 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.
- [9] 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.
- [10] Bourgeois, L.J., "Strategic Goals, Perceived Uncertainty, and Economic Performance in Volatile Environments", *Academy of Management Journal*, 28(3), 1985, pp. 548–573.
- [11] Burns, T.E. and G.M. Stalker, "The Management of Innovation", University of Illinois at Urbana-Champaign's Academy for Entrepreneurial Leadership Historical Research Reference in Entrepreneurship, 1961.
- [12] 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, pp. 263–290.
- [13] 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.
- [14] 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.
- [15] Fornell, C. and D.F. Larcker, "Evaluating Structural Equation Models with Unobservable Variables and Measurement Error", *Journal of Marketing Research*, 1981, pp. 39–50.
- [16] 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.
- [17] Gawer, A., "Platform Dynamics and Strategies: From Products to Services", in Gawer, A. (ed.) "Platforms,

- Markets and Innovation”, Edward Elgar, Cheltenham, 2009, p. 45-74.
- [18] 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.
- [19] 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.
- [20] Hoetker, G., "Do Modular Products Lead to Modular Organizations?", *Strategic Management Journal*, 27(6), 2006, pp. 501–518.
- [21] Kaplan, S. and B.J. Garrick, "On the Quantitative Definition of Risk", *Risk Analysis*, 1(1), 1981, pp. 11–27.
- [22] 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.
- [23] 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.
- [24] 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.
- [25] 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.
- [26] March, J.G. and Zur Shapira, "Managerial Perspectives on Risk and Risk Taking", *Management Science*, 33(11), 1987, pp. 1404–1418.
- [27] Meyer, A.D., A.S. Tsui, and C.R. Hinings, "Configurational Approaches to Organizational Analysis", *Academy of Management Journal*, 36(6), 1993, pp. 1175–1195.
- [28] Mikkola, J.H. and O. Gassmann, "Managing Modularity of Product Architectures: Toward an Integrated Theory", *IEEE Transactions on Engineering Management*, 50(2), 2003, pp. 204–218.
- [29] Milliken, F.J., "Three Types of Perceived Uncertainty about the Environment: State, Effect, and Response Uncertainty", *Academy of Management Review*, 12(1), 1987, pp. 133–143.
- [30] Nidumolu, S., "The Effect of Coordination and Uncertainty on Software Project Performance: Residual Performance Risk as an Intervening Variable", *Information Systems Research*, 6(3), 1995, pp. 191–219.
- [31] Nooteboom, B., H. Berger, and N.G. Noorderhaven, "Effects of Trust and Governance on Relational Risk", *Academy of Management Journal*, 40(2), 1997, pp. 308–338.
- [32] 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.
- [33] Ragin, C.C., "Redesigning Social Inquiry: Fuzzy Sets and Beyond", University of Chicago Press, Chicago, 2008.
- [34] Ragin, C.C., K.A. Drass, and S. Davey, "Fuzzy-set Qualitative Comparative Analysis 2.0", Tucson, Arizona: Department of Sociology, University of Arizona, 2007.
- [35] Raz, T., A.J. Shenhar, and D. Dvir, "Risk Management, Project Success, and Technological Uncertainty", *R&D Management*, 32(2), 2002, pp. 101–109.
- [36] Rindfleisch, A. and J.B. Heide, "Transaction Cost Analysis: Past, Present, and Future Applications", the *Journal of Marketing*, 1997, pp. 30–54.
- [37] Rochet, J. and J. Tirole, "Platform Competition in Two-sided Markets", *Journal of the European Economic Association*, 1(4), 2003, pp. 990–1029.
- [38] Ropponen, J. and K. Lyytinen, "Components of Software Development Risk: How to Address Them? A Project Manager Survey", *IEEE Transactions on Software Engineering*, 26(2), 2000, pp. 98–112.
- [39] 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.
- [40] Schneider, C.Q. and C. Wagemann, "Set-theoretic Methods for the Social Sciences: A Guide to Qualitative Comparative Analysis", Cambridge University Press, Cambridge, 2012.
- [41] Simon, H.A., "The Architecture of Complexity", *Proceedings of the American Philosophical Society*, 106(6), 1962, pp. 467–482.
- [42] Stump, R.L. and J.B. Heide, "Controlling Supplier Opportunism in Industrial Relationships", *Journal of Marketing Research*, 1996, pp. 431–441.
- [43] Tiwana, A., "Does Technological Modularity Substitute for Control? A Study of Alliance Performance in Software Outsourcing", *Strategic Management Journal*, 29(7), 2008, pp. 769–780.
- [44] Tiwana, A., "Evolutionary Competition in Platform Ecosystems", *Information Systems Research*, 26(2), 2015, pp. 266–281.
- [45] Tiwana, A., "Platform Desertion by App Developers", *Journal of Management Information Systems*, 32(4), 2015, pp. 40–77.
- [46] 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.
- [47] Walker, G. and D. Weber, "A Transaction Cost Approach to Make-or-buy Decisions", *Administrative Science Quarterly*, 1984, pp. 373–391.
- [48] Wallace, L., M. Keil, and A. Rai, "How Software Project Risk Affects Project Performance: An Investigation of the Dimensions of Risk and an Exploratory Model", *Decision Sciences*, 35(2), 2004, pp. 289–321.
- [49] Williamson, O.E., "The Economic Institutions of Capitalism", Simon and Schuster, 1985.
- [50] Yoo, Y., O. Henfridsson, and K. Lyytinen, "Research Commentary: The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research", *Information Systems Research*, 21(4), 2010, pp. 724–735.