Managing the Sources of Uncertainty: Matching Process and Context in Software Development

Alan MacCormack and Roberto Verganti

There is increasing interest in the literature about the notion of a contingent approach to product development process design. This interest stems from the realization that different types of projects carried out in different environments are likely to require quite different development processes if they are to be successful. Stated more formally, a contingent view implies that the performance impact of different development practices is likely to be mediated by the context in which those practices operate. This article provides evidence to support such a view.

Our work examines whether projects in which the development process matches the context achieve superior performance. We focus on two sources of uncertainty that generate challenges for project teams: platform uncertainty, reflecting the uncertainty generated by the amount of new design work that must be undertaken in a project; and market uncertainty, reflecting the uncertainty faced in determining customer requirements for the product under development. We develop hypotheses for how these sources of uncertainty are likely to influence the relationships between a number of specific development practices and performance. We then test these hypotheses using data from a sample of 29 Internet software development projects.

Our results provide evidence to support a contingent view of development process design. We show that in projects facing greater uncertainty, investments in architectural design, early technical feedback, and early market feedback have a stronger association with performance. The latter relationships are influenced by the specific sources from which this uncertainty stems: platform uncertainty mediating the impact of early technical feedback and market uncertainty mediating the impact of early market feedback. Our results also indicate that while greater uncertainty is associated with making later changes to a product’s design, this practice is not associated with performance.

Our findings suggest that managers carefully must evaluate both the levels and sources of uncertainty facing a project before designing the most appropriate process for its execution. In particular, they should explore the use of specific development practices based upon their usefulness in resolving the specific types of uncertainty faced. Importantly, these decisions must be made at the start of a project, with purposeful investments to create a process that best matches the context. Reacting to uncertainty ex-post, without such investments in place, is unlikely to prove a successful strategy.

Introduction

As we enter a new millennium, the management of innovation within firms is under increasing scrutiny. Despite a wealth of
research generating insights into the design of the new product development process, our knowledge of this process still seems to lag the requirements placed upon it by rapidly changing business environments. While many firms have implemented a range of formal processes through which development activities are conducted, anecdotal evidence suggests there is increasing dissatisfaction with these processes. It is our belief that this dissatisfaction arises not because the processes that have been implemented are designed poorly. Rather, the problem arises in the rigidity of their specification, which does not allow firms to tailor the specifics of the process to reflect the unique context facing each particular project.

The determinants of new product development success have been studied extensively in a stream of academic research over the past two decades [3,6,9,10,11,25,39]. Much of the early research was conducted in relatively stable industries, leading to the development of the stage-gate model for managing development. In this model, the process is divided into a number of separate stages, each of which is performed sequentially [2,7,26,41]. Milestones called gates mark transitions between stages, whereupon a review of progress occurs and a go/no-go decision is made with regard to continuing the project. While some evolutions of this model propose more “fuzzy” gates to separate stages and hence to allow concurrency [8], the basic principle remains the same: the freezing of the design as early as possible in development and the avoidance of changes to the design past this point.

Stage-gate models have been shown to work very well in the contexts in which they have been studied. Indeed, the uniform nature of the findings from such studies encourages a view that there might be “one best way” to organize the activities of product development (an appealing view, given the ease of implementing a one-size-fits-all solution). Yet such a theory of product development “best practice” would stand in the face of much research, which suggests that with regard to issues of organizational process and structure, we should adopt a “contingent” view [28]. Indeed, recent studies analyzing the performance of product development processes in uncertain and dynamic environments demonstrate that an effective process in these settings looks quite different from the stage-gate model [1,12,19,24,27,30]. Rather than a sequential process emphasizing planning and execution, successful firms tend to use a more iterative process, which emphasizes learning and adaptation. We call this a “flexible” process, where flexibility refers to the ability to generate and to respond to new information for a longer proportion of a development cycle.

The different models of development emerging from the streams of research described above are reconcilable if we view the design of the product development process as being contingent on the context in which that process operates. With such a view, firms should tailor the development process to reflect the context facing each specific project. This contingent perspective is gaining increasing attention in both the academic literature [24,30,31,34,45] and in ongoing research studies1. Most studies to date, however, merely describe how different projects follow different patterns in response to the environments they face. While there is some empirical evidence that in contexts with high levels of uncertainty, projects tend to make design changes at a later stage of a project [32], there has been little assessment of the impact these choices have on performance. The question remains, therefore, whether firms that match process to context actually

---

**BIOGRAPHICAL SKETCHES**

Alan MacCormack is the Lumry Family Assistant Professor of Business Administration at Harvard Business School. His research explores the design of the new product development process within high-technology firms, with a recent focus on the management of software development. His work has appeared in a number of books and journals, including *Harvard Business Review* and *Sloan Management Review*. He currently is teaching a new elective curriculum course on the management of technology-based ventures.

Roberto Verganti is professor of management of innovation at the Dipartimento di Ingegneria Gestionale di Politecnico di Milano. At Politecnico di Milano he is also director of the School of Doctoral Programs and co-director of the Master Degree in Strategic Design. His research interests are in the field of management of product innovation with a special focus on multimedia products and services and the Italian industrial design system. Professor Verganti is a member of the editorial board of the *Journal of Product Innovation Management* and is a member of the organizing committees of the EIASM International Product Development Management Conference and the EIASM Doctoral School in Technology Management.

---

1 An interesting perspective is provided by surveying three recent EIASM International Product Development Management Conferences in Como, Italy (1998), Cambridge, UK (1999), and Leuven, Belgium (2000). At these conferences, the number of articles specifically focused on managing contingencies were the following: five out of 74 articles (7 percent) in 1998 [16, pp. 65–74, 239–254, 419–434, 653–670, 763–776]; six out of 75 (8 percent) in 1999 [17, pp. 87–98, 145–156, 369–384, 547–580, 505–516, 667–678]; and eight out of 39 (20 percent) in 2000 [18, pp. 31–44, 75–90, 91–106, 189–203, 317–329, 347–368, 473–486, 551–573]. Despite this attention, however, only two of these articles examined the impact of such contingencies on project performance [33,37].
achieve better results, and if so, which specific development practices should be varied in response to which specific contextual factors.

This article examines whether projects that tailor the new product development process to reflect the specific uncertainties they face achieve superior performance. We investigate two sources of uncertainty that generate challenges for project teams: platform uncertainty, reflecting the amount of new design work which must be undertaken in a project; and market uncertainty, reflecting the uncertainty faced in determining the requirements that the resulting product must fulfill. Our analyses reveal that projects in which the development process “matches” the context faced indeed are associated with better performance. They illustrate that the association between different development practices and performance varies according to both the level of uncertainty facing a project as well as the specific sources of uncertainty.

In this article, first a conceptual framework for analyzing the interaction of process, context, and performance in the development of new products is introduced, leading to a number of hypotheses about the mediating effect of both platform and market uncertainty on the effectiveness of specific development practices. Then a description of the empirical design, which tests the validity of these hypotheses on a sample of completed projects in the Internet software industry, is given. The article concludes with a discussion of the results of the research.

Conceptual Framework

Figure 1 illustrates our research framework. At its heart is the view that in order to understand whether managing project contingencies can lead to competitive advantage in product development, one needs to consider three different factors: the development process adopted, the context faced, and resulting measures of project performance. This framework makes explicit the reciprocal role played by these factors.

It is important to note that most research in the field of new product development concentrates on what we call the “main effect” relationship in this framework—that is, the link between process choices and performance. There typically is little consideration given to the differing contexts that a project faces, the assumption being that selecting projects of a similar type within a single industry controls for such differences. Given that our objective here is to examine the impact of these contextual differences, however, we seek to do the opposite: to maximize variations in the context facing different projects in our sample.² For our purposes, we therefore focus on environments that are highly uncertain and dynamic. By uncertain, we mean that evolutions in customer needs and the technologies required to meet these needs are hard to predict; by dynamic, we mean that these evolutions occur rapidly. These types of environments are likely to exhibit greater diversity in terms of the contexts faced by individual projects, increasing the importance of an appropriate match between process and context. In stable environments, there is likely to be less variation in the context facing individual projects, hence the importance of matching process to context is less important.

With regard to measuring attributes of the context, we focus here on the sources of uncertainty affecting a project.³ Our assumption is that uncertainties that stem from different sources may require different responses in terms of development process design. With regard to measuring attributes of the development process that should be varied in response to these uncertainties, we draw upon previous studies of product development in highly uncertain and dynamic settings that highlight the importance of practices that support flexibility[34]. Hence, our research questions can be stated more precisely as follows: Do development practices that support a more flexible process have a stronger association with performance in projects that face more uncertain contexts? And if so, how do these relationships vary

² Subject to the restriction that we look at projects within the same industry. Comparing projects across different industries makes accurate performance comparisons difficult due to industry specific factors.

³ Other context variables might include, for example, differences in a firm’s country of location (e.g., culture, infrastructure, etc.), or differences in the specific nature of the technology (e.g., software versus hardware).
according to the specific sources from which this uncertainty stems?

**Defining Process Measures: Practices that Support Flexibility**

To understand how to achieve greater flexibility in development, we draw upon recent work investigating product development in uncertain and dynamic environments. In particular, we build on MacCormack et al.’s study of Internet software development, which describes the role of greater investments in architectural design, earlier feedback on a product’s system-level performance, and a project team with greater amounts of “generational” experience [34]. We focus here on the first two of these practices, as these represent parameters of the development process as opposed to a measure of the development team’s structure.

Greater investments in architectural design are required in a flexible process due to the creation of an additional design criterion for the product architecture—that of facilitating process flexibility. To achieve this objective, the architecture must support an early integration of the emerging product design (to gain feedback on performance at the “system” level) and also must have the ability to accept changes in functionality at a late stage of development (in response to newly emerging information). These requirements often are achieved by a more modular, or loosely coupled architecture (for example, the Linux operating system [35]). Such techniques, however, often conflict with the primary objective of architectural design activities—that of optimizing a product’s system-level performance [36,40,42,44]. The selection of the product architecture is therefore a more complex problem in projects facing greater uncertainty, requiring the allocation of additional resources (as compared to projects facing less uncertainty).

Early feedback on a product’s system-level performance is required in a flexible process to generate information on how well the emerging product design functions, both from a technical perspective and from the standpoint of how well it meets evolving market requirements. This feedback is critical given that in highly uncertain contexts, problematic interactions are likely to arise, both between a product’s component technologies (e.g., when developing a new architecture) as well as between the product and its operating environment (e.g., when developing a new piece of software for a desktop computer). Early feedback on system-level performance allows a team to gather information on the performance of a design and to identify problematic interactions among its constituent components. At the same time, it facilitates the gathering of information on new feature requests by customers, which might be implemented as part of the ongoing project.

We consider here one further measure of the development process often cited in the literature as a measure of flexibility: the extent to which a project makes major changes to the design at a late stage of development [23,30]. The motivation for including this measure relates to the advantages that a firm has if it can respond to new information late in a development cycle. In an uncertain environment, this allows a project team to deliver a superior product to customers by incorporating their latest requirements into the design. We therefore might expect that a measure of late design changes would be associated with better performance when faced with greater uncertainty. Note however, that the act of making late changes to a design is not a direct measure of flexibility but rather is an observed behavior. Projects that exhibit such behaviors may or may not have made the underlying investments necessary to cope with such changes. To the degree that they have not, such behaviors even might yield a negative relationship with performance. We will rely on our empirical analyses to shed light on this alternative hypothesis.

**Defining Context Measures: Sources of Uncertainty**

Many previous authors have identified uncertainty as an important influence on the appropriate structure for managing the activities of an organization [4,15,28,43]. However, authors often differ in their definitions of uncertainty and in their interpretations of the actions that are most appropriate when faced with great amounts of it (e.g., [38]). Indeed, there are likely to be several different sources of uncertainty that challenge development teams, each of which may require a different response in terms of optimal process design. In this article, we focus on two specific sources of uncertainty—platform uncertainty and market uncertainty.

Platform uncertainty is defined as the degree to which uncertainty exists over the specific design solutions that will be required in a project. This concept is related primarily to the extent of change in
the design vis-à-vis the previous version of the product (i.e., whether the product is a derivative product involving only minor changes to the existing design or is a major platform renewal that involves reexamining the architecture of the product [46]). The greatest degree of platform uncertainty occurs in “breakthrough” projects in which a firm has no prior experience or existing design upon which to build. By contrast, projects in which only a small amount of the existing design is changed present the least amount of platform uncertainty to development teams.

Market uncertainty is defined as the level of uncertainty that exists in the external environment, with regard to determining the requirements that customers have of the resulting product. Market uncertainty is high, for example, when customers have had little consumption experience with a product, thus making requirements difficult to define ex-ante (e.g., the first Internet browser). As we define it, market uncertainty relates to information that is not “knowable” at the outset of a project; hence, all firms developing products that target the same market at the same point in time will face a similar level of market uncertainty.

These definitions illustrate that the context facing a project is a complex mix of both exogenous uncertainties associated with the external environment and uncertainties, which are to some degree endogenous to a project (e.g., the choice of how much new design work to undertake). For our purposes however, we assume that the choices a development team makes that influence these uncertainties are determined prior to selecting the product development process that is to be used. As a result, we consider both sources of uncertainty as exogenous factors to which a response must be formulated in terms of development process design. Note also that with the definitions as stated, there is no reason to expect a correlation between the two sources of uncertainty. It is possible for projects to be low in platform uncertainty and high in market uncertainty (e.g., a derivative release of an Internet software product) and vice versa (e.g., a new platform for a coffee maker). The appropriate responses in terms of development process design therefore are assumed to be separable for the purposes of analysis.

Matching Process and Context

Here hypotheses are developed for how each source of uncertainty impacts the association between different development practices and performance.

The impact of platform uncertainty. Platform uncertainty is related primarily to the extent of change in the product design vis-à-vis the previous version of the product. Wheelwright and Clark, for example, describe how projects can be classified as derivative, platform, or breakthrough projects depending upon the degree of change required in the product and/or process design [46]. Projects that encompass significant changes to an existing design generally involve reexamining system-level (architectural) design choices, introducing greater uncertainty than choices that are made within the confines of the existing product architecture [36]. Greater platform uncertainty in a project, considered independently of the uncertainty stemming from other sources, therefore is likely to require a more flexible process, given that the firm is unfamiliar with the specific design solutions that must be adopted. In particular, given this source of uncertainty requires revisiting decisions, which are made at a higher-level in the design hierarchy, we would expect that investments in architectural design would be of greater importance [5]. Thus, Hypothesis 1 states the following:

H1: In projects that face greater platform uncertainty, investments in architectural design will have a stronger association with performance.

We should note at this point the precise wording of this hypothesis. While investments in architectural design may be associated with performance across all projects, we predict that the relationship will be stronger among projects that face higher platform uncertainty. That is, the relationship between process and performance is mediated by the context faced. Subsequent hypotheses will follow the same logical construction.

The fact that higher-level decisions must be revisited in a project facing greater platform uncertainty increases the importance of early feedback on a product’s system-level performance, given the increased likelihood of problematic interactions among parts of the design in a new architecture. In this case, the specific nature of the feedback required is concerned mainly with interactions between a product’s component technologies (i.e., it is not a response to market uncertainty). We therefore can be precise in stating the type of feedback, in Hypothesis 2, as follows:

H2: In projects that face greater platform uncertainty, early technical feedback on system-level performance will have a stronger association with performance.
Greater platform uncertainty in a project heightens the probability that changes to the system-level design will be required at a late stage of a project, given the additional uncertainty generated when changes are made to an existing product architecture. To the degree that the development process can accommodate such late changes, specifically at the architectural level, we expect that a project facing greater platform uncertainty would achieve superior performance. Thus, Hypothesis 3 states the following:

\[ H3: \text{In projects that face greater platform uncertainty, making major changes to the product architecture at a later stage will have a stronger association with performance.} \]

The impact of market uncertainty. Greater market uncertainty in a project, considered independently of the uncertainty stemming from other sources, is likely to require a more flexible process, given the need for a firm to respond to new information that arises during development. In particular, we expect that investments in architectural design would have increasing importance as the level of market uncertainty increases, given the need for an architecture that can facilitate ongoing changes to the product’s functionality. While this prediction mirrors that for platform uncertainty, it stems from a different dynamic. Investments in architectural design are useful in projects facing greater platform uncertainty due to the need to resolve decisions related to higher-levels of the design hierarchy (i.e., the product architecture). When faced with greater market uncertainty, however, these investments are useful to the degree that they provide an architecture that allows a rapid response to newly emerging requirements. As a result, even in derivative projects (where the product architecture does not change significantly) we expect investments in architectural design to have increasing importance when faced with greater market uncertainty. Thus,

\[ H4: \text{In projects that face greater market uncertainty, investments in architectural design will have a stronger association with performance.} \]

The presence of greater market uncertainty in a project is likely to increase the importance of early feedback on a product’s system-level performance. In contrast to projects that face greater platform uncertainty, however, the need for feedback is concerned not so much with the interactions between a product’s technical components but rather is concerned with the product’s performance in the end-use context, given this is where the source of uncertainty lies. We therefore discriminate between feedback mechanisms that are primarily internal by nature (i.e., technical feedback) and those that focus on external validation (i.e., market feedback). Thus,

\[ H5: \text{In projects that face greater market uncertainty, early market feedback on system-level performance will have a stronger association with performance.} \]

Greater market uncertainty in a project increases the probability that design changes are required late in a project to respond to new information on customer requirements. To the degree that the development process can accommodate late changes to the design, we expect this would result in a product that more closely matches customer needs. The nature of these design changes, however, is likely to be different than the changes brought about as a result of greater platform uncertainty, which relate to evolutions in the product architecture. Rather, greater market uncertainty increases the probability that changes are required at the “feature” level, as customer requirements for the product are clarified through feedback from the market. Thus, Hypothesis 6 states the following:

\[ H6: \text{In projects that face greater market uncertainty, making major changes to the product’s features at a later stage will have a stronger association with performance.} \]

Figure 2 summarizes the hypotheses discussed above.

**Empirical Design**

The empirical analysis is based on data captured during a study of product development practices in the Internet software industry from 1996 to 1998 [31]. During the study period, extreme market uncertainty existed in this industry. Many customers lacked even a basic knowledge of what was possible with web-based technologies. Yet while the overall level of market uncertainty was high, there was still significant variation in the relative levels from project to project. For example, one product in our sample is a software tool that helps create business applications for the web by linking internal company databases to a website. Although its commercial impact is significant, the customer requirements it fulfills can be traced to previously known contexts in terms of...
management information systems. In other applications, however, the changes were more dramatic. For example, web browsers were a product category that represented a completely new context of use for customers. Understanding requirements for these types of products and being able to follow them as they evolved during development therefore presented a much greater challenge to project teams.

In addition to variations in market uncertainty, the industry exhibited significant variations in platform uncertainty during the study period. At any one time, pioneering entrants may have been developing their second or third release of a product, while latecomers were developing their first. For example, in the summer of 1996, Netscape was developing the third release of its “Navigator” browser, which added incremental improvements to their existing product, while Microsoft was engaged in developing the first major platform release of its “Explorer” browser.4

Research was completed in two stages: In the first, interviews were conducted with project managers at several companies both to understand the specific nature of the development process in this industry and to identify how to operationalize measures of process and context; In the second, data were collected through the use of a survey instrument distributed to project managers. In most cases, we also visited managers to collect additional descriptive data. We distributed the survey to 39 firms, of which 17 provided complete data on one or more projects, a response rate of 43.6 percent. Our final sample consists of 29 completed projects.

The Measure of Performance

To measure performance, we captured the relative quality of the final products produced, controlling for the resources allocated to each project. Given the specific nature of the industry and its products (e.g., the fact many products are distributed for free or are bundled with other products) we focused on product and project level measures rather than on measures of the resulting financial performance (e.g., [21,22]).

Product quality. To evaluate the relative quality of products in our sample, we relied upon the assessments of a panel of experts gathered using a two-round Delphi process [13]. The panel of experts consisted of 14 industry observers from a variety of journals and websites that review Internet software products. Experts were asked to rate the “overall quality of products relative to other competitive products that targeted similar customer needs at the time the product was launched.” Overall quality was defined as a combination of performance, functionality, and reliability. Assessments were given on a seven-point Likert scale, with 4 indicating that the product was at parity with competitive offerings. Experts were asked to indicate their level of knowledge of the product (on a 1–7 scale) and to list up to three competitive products against which their comparisons were made. Experts then were provided with summary information on data from the first round and were given an opportunity to update their scores. After the second round, the Delphi process had converged. An analysis of the standard deviation of experts’ assessments in the second round demonstrated that the highest variance in experts’ judgments

4 The previous two versions of Explorer had been based upon technology procured from Spyglass.
of product quality came from experts who rated their degree of confidence equal to 3 or less. Hence we use the mean score provided by knowledgeable experts (i.e., those scoring 4 or greater) as our dependent variable.

**Control variable: resources.** We captured a measure of the development and test resources (in person-days) allocated to each project to control for the potential impact of resources on product quality. We adjusted this measure to reflect the fact that the complexity of products in the sample varies, modeling the resources allocated to each project as a function of both the lines of new code developed in each project and a dummy that indicates whether the project was to develop a new service (see [34]). The dummy was necessary because projects to develop new services (e.g., websites) were observed to achieve higher productivity rates than other projects in terms of lines of code per person-day due to the use of different programming languages and instructions. The residuals from this model are used as our control variable.

**Measures of the Development Process: Practices that Support Flexibility**

Five measures of the development process associated with a more flexible process were captured: investments in architectural design, early technical feedback, early market feedback, late changes to a product’s architecture, and late changes to an individual module.

**Investments in architectural design.** In Internet software development projects, major efforts are required in architectural design activities. The product architecture must on the one hand optimize the operating performance of the product (e.g., in terms of memory use or speed), yet on the other hand must facilitate process flexibility (e.g., through the ability to make changes rapidly to the existing design). The emphasis a project places on resolving these potentially conflicting objectives was captured by calculating a ratio of the resources allocated to architectural design activities relative to those allocated to development and test. Given the size of products in the sample varies significantly, this ratio was adjusted to control for scale effects.

**Early technical and market feedback.** In Internet software development projects, early feedback on a product’s system-level performance is essential in solving emerging technical challenges as well as in understanding how the product performs in the end-use context. In projects in this industry, two milestones stand out as indicators of how early this feedback occurs: the point at which a product’s modules first are integrated into a working version of the system; and the point at which the first beta release of this system is distributed to external customers (see Figure 3). The first system integration occurs partway through the detailed design stage, representing the first point at which the project team receives technical feedback on how modules in the design interact. The first beta release occurs sometime after this, representing the first time that the project team receives market feedback on the functioning of the product in the end-use context (although the product typically is not complete at this point). While projects sometimes release multiple beta versions prior to product launch, the most significant milestone is the first release, which marks the start of the period when customers can begin giving feedback on the emerging design.

The way the points at which the first system integration and first beta release occur are operationalized is to capture the percentage of the product’s final functionality developed at each milestone (the functionality contained in the product is defined as 0 percent when the project starts and 100 percent at launch)\(^6\). A lower figure therefore indicates that feedback is received earlier. Given the size of products varies significantly, we control these measures for scale effects prior to use.

**Late changes to the product architecture and to an individual module.** In Internet software development projects, the ability to make major changes to the design at a late stage of development allows a team to mirror evolutions in technology and market requirements as a project proceeds. To capture the extent to which projects in our sample respond to such information, we consider two additional milestones in development: the point at which the last major change is made to the product architecture; and the point at which the last major change is made to an individual module. The former provides a perspective on how late a team is able to introduce deep changes in the product’s structure before launch. The latter

---

5 To control for scale effects, our raw process measures are modeled as a function of the logarithm of new code developed, and the residuals from this model are used in the analyses. This approach is published in [34].

6 Using dates to measure how early these milestones occur is problematic, given that delays just prior to shipping can skew the outcomes (see [34]). Capturing the functionality at each milestone is a more robust measure.
provides a perspective on how late in the project the team is able to introduce major changes to a product’s features to meet customer requirements better. A product’s “architecture” is defined as the arrangement of functional requirements, the allocation of functional requirements to specific modules, and the design of interfaces between these modules (derived from [44]). A change to a module is defined as one that does not affect other modules [44]. Our fieldwork indicated that changes to a product’s features involved changes at the module level (i.e., the overall product architecture was not affected). These measures are operationalized by capturing the time between the last major change to the design (at both the architecture and the module level) and the product launch. To account for the fact that the duration of projects varies significantly, this figure was divided by the overall cycle-time of the project (i.e., the time between the start of concept design and product launch). The resulting measure captures the proportion of a project’s development cycle in which the design is “frozen.” A lower figure therefore indicates the design was frozen later (i.e., late changes were made to the design).

**Measures of Project Context: The Sources of Uncertainty**

**Platform uncertainty.** We measure platform uncertainty by capturing the percentage of new code developed in the final product. This provides a direct measure of the amount of new design problems that the project team has to solve compared to solutions that are carried over from previous product releases. It also reflects the extent to which the desired innovation affects the overall and inner structure of the product (i.e., the extent to which a new product architecture is required). Note that managers also were asked to classify the project either as a platform release (i.e., a project that requires a new product architecture) or as a derivative release (i.e., a project focused only on incremental improvements to an existing design). The strong correlation between these assessments and the percentage of new code ($\rho=0.717$, $p<0.01\%$) confirms that the latter is a good surrogate for the degree of change in the product architecture.

**Market uncertainty.** To assess the level of market uncertainty facing each project, we exploit the data on product quality gathered in the Delphi process. In particular, we assume that the uncertainty in customer requirements facing development teams is reflected by the residual variation among experts’ judgements of product performance. Variation among experts’ assessments of the same product after the first round of the Delphi can be attributed to several factors: a lack of knowledge of the product; a different understanding of the market segment in which the product competes; ambiguity in the scale for assessment; and actual variation in experts’ judgments. The second round of the Delphi attempts to eliminate variability due to the first three of these factors. In our sample, this generated a convergence in experts’ opinions on individual products, while creating additional variance in the spread of mean performance outcomes.
across products. A third round with two pilot products showed that experts were not willing to change their assessments further and that iteration of the process would not generate additional convergence [29]. This residual deviation therefore reflects actual variations in experts’ subjective judgments of product quality. Our assumption is that this variation provides a proxy for the uncertainty faced by the project team in determining customers’ requirements for the product. That is, if there were no uncertainty in determining what constitutes a high-quality product, all experts ultimately would converge to the same assessment. Hence, we use the mean absolute deviation of experts’ assessments for each product remaining after the second round of the Delphi as our measure of market uncertainty. (For a similar application of measuring uncertainty by use of the variation in experts’ judgments, see [20].)

Descriptive Statistics

Table 1 summarizes the data on our sample of projects. To make the table more meaningful, measures of the development process are reported with their original values rather than with their values after adjusting for scale effects. Data also are reported on the size of the resulting products.

The Appendix reports correlations between the final measures used in our analyses. Several points should be highlighted. First, note that the measures of platform uncertainty and market uncertainty are not correlated, confirming that these represent orthogonal sources of uncertainty that must be dealt with by development teams. Second, note that each of these measures of uncertainty has no correlation with product quality. This is consistent with the notion that even in projects that face great uncertainty, performance may not suffer, providing an appropriate process is adopted. Third, note that the only practices that are correlated with either measure of uncertainty concern the introduction of late changes to the design. The results suggest that projects facing higher platform uncertainty tend to make later changes to the product architecture and that projects facing higher market uncertainty tend to make later changes to individual modules. None of the other practices supporting a more flexible process are correlated with either source of uncertainty. The question remains however, as to whether those projects in which there is a match between development process and context actually achieve superior performance. This question is examined next.

Analysis

The modeling approach involves the use of ordinary least-squares (OLS) regression to examine the effect of process and context measures on product quality (as rated by knowledgeable experts) using a control for resources. We use a stepwise-forward procedure, which is preferable to stepwise-backward regression given the small sample size [14]. This procedure has been adopted in similar studies [6,34]. We report the final models for three separate analyses: The first includes only the direct effects of process variables; the second includes both the direct effects and their interaction with platform uncertainty; and the third includes both the direct effects and their interaction with market uncertainty (see Table 2).

Model 1 describes the final model for product quality including only measures of the development process and a control for resources. This model does not account for the context each project faces, testing only for the overall importance of these process variables in this sample of projects. In this model, two variables are significant: architectural design effort and early market feedback. In total, these variables explain 36.6 percent of the variance in product quality. Surprisingly, we find that early technical feedback does not appear in this model, even though it is correlated with product quality (see the Appendix). The reason is that early technical feedback is correlated with early market feedback, which has a stronger relationship with performance. This correlation is an artifact of the way these measures are defined, given that market feedback cannot occur earlier than technical feedback (i.e., an early system integration is a prerequisite for an early beta release). We therefore conclude that some of the explanatory power of early market feedback likely is due to the practice of an early system integration, with the remainder being due to the choice of whether subsequently to release this early version to customers, an issue examined further below. Finally, we note that the two variables measuring the extent to

---

7 After the first round of the Delphi, the average quality of products ranged from 4.00 to 5.58, with a standard deviation of 0.45. The mean standard deviation of experts’ assessments for individual products was 0.87. After the second round, the average quality of products ranged from 3.5 to 5.52, with a standard deviation of 0.49. The mean standard deviation of experts’ assessments for individual products dropped to 0.57.
which late changes are made to the design are not significant. We revisit this result later, after commenting on more complex models combining both process and context variables.

Model 2 describes the final model for product quality including measures of the development process and their interaction with platform uncertainty. In this model, the interaction between...
architecture design effort and uncertainty is significant, providing evidence to support H1. Specifically, in projects that face greater platform uncertainty, investments in architectural design have a stronger association with performance. None of the other interaction terms examined appears as significant in this model.

Model 3 describes the final model for product quality including measures of the development process and their interaction with market uncertainty. In this model, the interaction between architectural design effort and uncertainty is again significant, providing evidence to support H4. Specifically, in projects that face greater market uncertainty, investments in architectural design have a stronger association with performance. None of the other interaction terms examined appear as significant in this model.

The aforementioned results indicate the importance of investments in architectural design activities when faced with uncertain contexts. As suggested in H1 and H4, the association between these investments and performance increases as both platform and market uncertainty increase. The analysis of early technical and early market feedback did not provide evidence to support the view that there is an interaction between these development practices and a project’s context. Unfortunately, however, the correlations that exist between several independent variables make it difficult to interpret these results conclusively with this modeling approach (see the Appendix). We therefore conducted further analyses to gain insights into the relationships suggested in H2 and H5. Specifically, we twice split the sample into subsamples, the first discriminating between high and low levels of market uncertainty and the second discriminating between high and low levels of platform uncertainty. The analysis of early technical and early market feedback did not provide evidence to support the view that there is an interaction between these development practices and a project’s context. Unfortunately, however, the correlations that exist between several independent variables make it difficult to interpret these results conclusively with this modeling approach (see the Appendix). We therefore conducted further analyses to gain insights into the relationships suggested in H2 and H5. Specifically, we twice split the sample into subsamples, the first discriminating between high and low levels of platform uncertainty and the second discriminating between high and low levels of market uncertainty. We then ran the main effects models for both early technical feedback and early market feedback on each subsample (see Table 3).

Models 4 and 5 indicate that early technical feedback has no association with performance in the subsample with low platform uncertainty but has a strong association with performance in the subsample with high platform uncertainty. Models 6 and 7 indicate that early market feedback has no association with performance in the subsample with low market uncertainty but has a strong association with performance in the subsample with high market uncertainty. The results are consistent with H2 and H5 developed earlier. They provide evidence that the source of uncertainty is an important dimension of a project’s context in terms of the mediating effect it has on specific development practices.

As an illustration of these dynamics, we show in Figure 4 a scatter plot of the relationship between early market feedback and market uncertainty (the association with performance is shown by differentiating the most and least successful projects). We notice that in projects facing low market uncertainty (i.e., where the mean average deviation is below 0.5), projects tend to adopt similar processes in terms of how early they release a beta to customers. The variation in development approach is much greater, however, in projects facing high market uncertainty. It appears that in projects that face extremely uncertain markets, teams tend to adopt extreme behaviors: Some release the first beta version significantly earlier than the mean, while others tend to release the first beta significantly later. A look at the performance of these projects indicates that the former achieve better results.

Our final comment concerns the ability to make major changes to the design at a late stage of a project. Our results suggest that the introduction of late changes to the product design (whether directed at the architectural level or at the level of individual modules) is not associated with performance, even when controlling for a project’s context. We therefore reject H3 and H6. The results stand in contrast to other research that suggests late design changes are a desirable attribute when a project faces great uncertainty (e.g., [24]). Our interpretation of these findings is that late changes to the product design are not an indicator of flexibility per se. In fact, they are likely to be made by any development team operating in an uncertain environment, regardless of whether the team has made the necessary investments to cope with such changes. In support of this view, consider that our measures capturing late changes to

---

8 To split the sample, we used the mean as a dividing line between high and low levels of uncertainty.
9 This technique reduces the available degrees of freedom; hence, care must be taken in interpreting the results.

10 To assess relative performance, we use the residuals from a regression model examining the effect of resources on product quality.
11 A similar dynamic, albeit less marked, holds for early technical feedback and platform uncertainty.
12 No further analyses were done with these variables given that their main effects were not correlated with performance and that they had no correlation with other variables that might cloud their effect in more complex models.
the product design are correlated with uncertainty. Specifically, platform uncertainty is correlated with later changes to the product architecture, and market uncertainty is correlated with later changes to an individual module. Projects facing high uncertainty therefore do tend to make changes to the design at a later stage of development. Critically, however, these changes appear as likely to lower performance as they are to improve it significantly.13

### Table 3. Effects of Process Measures on Product Quality in Subsamples

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant</td>
<td>4.351****</td>
<td>4.587****</td>
<td>4.466****</td>
<td>4.414****</td>
</tr>
<tr>
<td>(0.131)</td>
<td>(0.089)</td>
<td>(0.103)</td>
<td>(0.097)</td>
<td></td>
</tr>
<tr>
<td>Control: Resources</td>
<td>0.104</td>
<td>-0.009</td>
<td>0.093</td>
<td>0.151</td>
</tr>
<tr>
<td>(t=0.43)</td>
<td>(t=-0.03)</td>
<td>(t=0.45)</td>
<td>(t=0.85)</td>
<td></td>
</tr>
<tr>
<td>Early Technical Feedback</td>
<td>-0.015**</td>
<td>-0.007</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>(0.007)</td>
<td>(0.007)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Early Market Feedback</td>
<td>N/A</td>
<td>N/A</td>
<td>-0.026****</td>
<td>-0.005</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(0.007)</td>
<td>(0.008)</td>
</tr>
<tr>
<td>R-squared (adj)</td>
<td>17.1%</td>
<td>0.0%</td>
<td>57.2%</td>
<td>0.0%</td>
</tr>
<tr>
<td>F-ratio</td>
<td>2.34</td>
<td>0.60</td>
<td>10.3</td>
<td>0.58</td>
</tr>
<tr>
<td>Df</td>
<td>11</td>
<td>12</td>
<td>12</td>
<td>11</td>
</tr>
</tbody>
</table>

* p<10%  
** p<5%  
*** p<1%  
**** p<0.1%  
NB: Unless stated, numbers in brackets are standard errors.  
a Adjusted for product complexity.  
b Adjusted for scale effect.

### Figure 4. The relationship between early market feedback and market uncertainty

Implications and Conclusions

For practitioners, our findings have major implications, both in terms of the specific practices and uncertainties investigated here and in their broader connotations for the design of development processes in general. At a specific level, our results suggest that when faced with little or no uncertainty in terms of the platform being developed or the market being targeted, a traditional stage-gate process is likely to be effective. For example, there is little need to focus on an early system integration and subsequent product distribution to beta customers, given that the expected benefits from such feedback in terms of new information are low. This is especially true to the
degree that there are costs to such practices, which, as we pointed out earlier, there often are (e.g., a more modular product architecture typically has lower performance than a tightly coupled architecture [44]). In essence, early feedback provides an “option” to respond to new information, but the value of this option must exceed the cost of its acquisition to be worthwhile. When a project faces greater uncertainty, this is more likely to be true.

Importantly, our results also suggest that when faced with greater uncertainty in development, managers must avoid the pitfall of thinking that flexibility can be achieved merely by reacting to new information, for example, in making changes to the design at a late stage of a project. While such behaviors convey the appearance of flexibility, they are not, in fact, associated with performance. Our results suggest instead, that the roots of a flexible process are built in the early stages of a project, through activities such as investments in architectural design, which are geared to creating a design that can evolve as a project progresses. Only those projects in which such purposeful investments have been made are likely to benefit from late design changes, given the problems such changes can bring in a process that has not been designed with flexibility in mind.

While the results discussed above yield useful insights for practitioners, it is at the conceptual level that our work has the most powerful implications. Specifically, our findings lend support to a contingent view of the new product development process, suggesting that managers must avoid the potential pitfalls of “best practice” thinking and instead must evaluate a wide range of contextual factors before deciding upon the most appropriate process to adopt in any given project. In doing this, they need to address the levels of uncertainty faced, the specific sources from which this uncertainty stems, and the specific practices that are used best to resolve these uncertainties. Such a view implies that the first stage of any project should not, in fact, be concerned with the product design, but rather should focus on the design of the development process itself.

How then does one address concerns that designing a formal development process is an arduous and lengthy task that possibly cannot be undertaken within each project? One approach that emerged from our fieldwork was to identify the major types of uncertainty affecting a firm’s projects up front and then to define how to respond to different levels of each. For example, using the two types of uncertainty we describe above with two levels of each (high and low) would require four different process types. A stage-gate process might be defined for derivative projects facing low market uncertainty. At the other end of the spectrum, a highly flexible process might be defined for a platform project facing high market uncertainty. Identifying the drivers for these different processes and drawing up templates for each type would be an excellent first step along the road to matching process and context.

For the academy, this article provides evidence that studies investigating contingent approaches to product development necessarily must consider three different factors: the development process, the project context, and the resulting measures of performance. To illustrate, if this study had focused only on the relationship between process and context, we would have concluded that when faced with highly uncertain contexts, projects tend to make changes to the design at later stages of development. However, we could not have shown that these behaviors are not associated with performance, even when controlling for context. Conversely, we would not have detected any contingent behaviors associated with investments in architectural design or early feedback on product performance, given these have no correlation with either measure of uncertainty. Yet when we include the measure of performance, a contingency does emerge: Each of these behaviors has a stronger association with performance when faced with specific sources of uncertainty. We conclude that without careful consideration of process, context, and performance, research into contingent practices is likely to reveal only half the story.

The authors thank Rocco Mosconi for his helpful suggestions on the analysis; Abbie Griffin for her insightful comments on the manuscript; and two anonymous reviewers. We also thank the experts who helped assess the quality of products: Keith Allison, Whit Andrews, Christofer Barr, Rik Drummond, James Farley, Mark Frauenfelder, Carlo Ghezzi, Logan Harbaugh, Justin Hibbard, Kevin Jones, Eric Lundquist, Melanie McMullen, Shawn Rogers, and Gus Venditto. Financial support from the Consiglio Nazionale delle Ricerche (CNR) funds “Approaches and methods for risk management in the Internet products and services industry” gratefully is acknowledged.

References


37. de Moorloose, C. “New product success factors, new product typology, and moderating impact on success factors according to the type of newness,” EIASM 7th International Product Development Management Conference, Leuven, Belgium (May 2000).


## Appendix. Correlation Table

<table>
<thead>
<tr>
<th></th>
<th>Quality</th>
<th>Resources</th>
<th>Architect. Design Effort</th>
<th>Early Tech.: First Integration</th>
<th>Early Market: First Beta</th>
<th>Last Change to Architect.</th>
<th>Last Change to Module</th>
<th>Platform Uncertainty</th>
<th>Market Uncertainty</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quality</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Resources</td>
<td>0.025</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Architecture</td>
<td>0.431**</td>
<td>−0.323</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Early Tech.: First Integration</td>
<td>−0.423**</td>
<td>0.294</td>
<td>−0.336*</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Early Market: First Beta</td>
<td>−0.636***</td>
<td>0.132</td>
<td>−0.279</td>
<td>0.481***</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Last Change to Architecture</td>
<td>−0.083</td>
<td>−0.069</td>
<td>−0.229</td>
<td>−0.277</td>
<td>0.357*</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Last Change to Module</td>
<td>−0.060</td>
<td>0.264</td>
<td>0.093</td>
<td>0.379**</td>
<td>0.164</td>
<td>−0.011</td>
<td>1.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Platform Uncertainty</td>
<td>−0.195</td>
<td>−0.128</td>
<td>0.188</td>
<td>0.207</td>
<td>0.002</td>
<td>−0.500***</td>
<td>−0.183</td>
<td>1.000</td>
<td></td>
</tr>
<tr>
<td>Market Uncertainty</td>
<td>−0.061</td>
<td>0.115</td>
<td>0.104</td>
<td>−0.308</td>
<td>0.052</td>
<td>0.185</td>
<td>−0.456**</td>
<td>0.327</td>
<td>1.000</td>
</tr>
</tbody>
</table>

*p < 10%

**p < 5%

***p < 1%