On the nature of Over Featuring in the New Product Development process

Authors

Giacomo Marzi, U. of Trieste, Italy, giacomomrz@gmail.com

"Beyond Pathologies" in New Product Development

Abstract

The tendency to develop new products and services beyond what is required by market, users, plans, and the resources in the organization ranks at top 10 of risks of new product development (NPD) failures. However, scholars and practitioners devoted limited attention to it. Many terms are used to outline the excesses in product development such as Feature Creep, Feature Fatigue, Overdesign, Overspecification, Over-requirement, Scope Creep, and Gold-Plating among the others. Excessive development configures in various forms that can be grouped inside the name of "Beyond Pathologies" (BPs). Nevertheless, a terminological and conceptual confusion exist inside the realm of BPs together with a lack of theoretical development, and a limited investigation of the causes, the drivers, and the performance effects of BPs. The present paper presents an overview of the of BPs, the evolution of the concept over time, together with a meta-synthesis of findings emerged from the available empirical studies about BPs on the NPD projects. In doing so, a theoretical and conceptual development of BPs is proposed to clarify and advance such a multifaced phenomenon. BPs are also explored inside the most common NPD frameworks, namely Stage-Gate and Agile.

1. Introduction

Creating new-to-world products, offering new services, and continuously innovating the products portfolio, allows companies to get a better position on the market and beat the increasing rivalry (Christensen and Bower, 1996; Marzi et al., 2021; Wouters et al., 2011). However, the critical task of creating a competitive position though developing something new lies in the effective management of uncertainty and fuzziness that is inherent into the innovation process (Christensen and Bower, 1996; Yu et al., 2010). Uncertainty comes from both exogenous and endogenous sources, playing a crucial role in the success of the innovation and development process. The first, such as shifting customer preferences, competitors' strategic moves, and nascent technological trajectories. The latter such as the developers' creative responses to discoveries during the project or the co-evolution of technical solutions in the interacting product components (Antioco et al., 2008; Burke, 2013; Ren and Zhao, 2021; Wouters et al., 2011). As a result, managers, project managers, engineers, and developers have to make decisions with scarce information, high ambiguity, a vague overview of the market needs, overestimations, and judgmental biases that could favour the emergence of several pathological states of the NPD projects (Antioco et al., 2008; Burke, 2013; Coman and Ronen, 2010; Elliott, 2007; Yu et al., 2010).

At the same time, the increasing need of products' distinctiveness, the continuous technological developments, and the rapid changes in customers preferences pushed companies to develop products and services with alluring characteristics able to seduce the consumers and offer performances and features beyond what is really and needed by the users in the everyday usage[1] (Antioco et al., 2008; Burke, 2013; Coman and Ronen, 2010; Mikkola, 2003; J. B. Schmidt and Calantone, 2002; Thompson et al., 2005; Wouters et al., 2011).

[1] The so-called "seduction of excess" is a human legacy question. The well-known Latin expression proposed by the poet Horace "est modus in rebus" prompts for the need of wise moderation and sense of measure to avoid any type of excess. Specifically, Horace warns by taking care not to run into too much or too little and adopting

Starting from this premises, the present study defines and classifies as "Beyond Pathologies" a set of uncertainty-related tendencies that can seriously harm the success of an NPD process, spanning across the fields of innovation management, R&D management, engineering, and design.

The term "Beyond Pathologies", from now "BPs", comprehends all the conditions when a product or a service is developed beyond what is needed by the users, by the market, by the plans, and beyond what is feasible with the company's resources (Bianchi et al., 2019; Bjarnason et al., 2012; Buschmann, 2009, 2010; Coman and Ronen, 2010; Shmueli and Ronen, 2017; Thompson et al., 2005).

BPs could manifest in different forms and at different levels, by a deliberately choice of the NPD team or as a result of irrational behaviour. BPs comprehend a wide set of sub-pathologies related to uncertainty, from the steady increase of the project's scope (Scope Creep) to continuous inflow of additional features when the product is still in development (Overdesign and Feature Creep). The consequences of BPs are numerous, often becoming a steep price to pay. Project delays, budget overruns and users' difficulty in use excessively features product are common outcomes in several NPD projects (Rust et al., 2006; Stock, 2011; Thompson et al., 2005; Verkijika, 2021).

The magnitude of risk associated with BPs is so pervasive that even the NASA listed the inclusion of excessive features among the top 10 risks of failure for development projects (Landis et al., 1992, pp. 98–128). Several sound examples of the effects of BPs on NPD projects are available in the everyday experience. BMW series 7 included the iDrive system, which proposed about 700 capabilities requiring multifunction displays and multi-step operations. Because of the complexity of iDrive system, BMW was forced to include an

---

the right equilibrium required by the situation. The complete sentence of Horace is "Est modus in rebus sunt certi denique fines, quos ultra citraque nequit consistere rectum" that can be translated as "There is an optimal condition in all things with precise boundaries beyond which one cannot find the right thing". The sentence can be found in Horace, Satire (1, 1, 106–107).

instruction manual, which was thought as necessary whether a valet parker took the car (Rust et al., 2006). Mercedes-Benz removed around 600 non-essential functions from its cars as they were the direct cause of several malfunctions to electronic parts, lack of usability, and increased need for after-sale support (Rust et al., 2006). From 2010, Apple included in its product the Retina display to attract new customers and showcase a distinguish feature, claiming that Retina has more resolution than the human eyes can perceive (Edwards, 2010). Although it has been an impulse to innovation in the entire display industry, a resolution beyond what is perceivable from human eyes brought several inconvenient, from the higher costs and complexity in the devices' architecture, to the tangible battery drain leading to worse usability coming from decreased equipment' battery life (Edwards, 2010; Liu and Yu, 2017).

The consensus around that excessive complexity of products damage the performance in terms of longer development cycles, worst user experience, worst ergonomics, decrease usability, and increased need of after-sales service to support more defective products is clear (Backman et al., 2007; Coman and Ronen, 2010; Mitchener, 2008; Rust et al., 2006; Thompson et al., 2005; Verkijika, 2021; Wouters et al., 2011). However, despite the awareness about the magnitude of BPs, the topic received a scarce attention, especially outside the software development domain (Shmueli and Ronen, 2017) while only recent literature highlighted the relevance of BPs on physical products and services (Cesaretto et al., 2021; De Giovanni, 2019; Gregori and Marcone, 2019; Jain, 2019; Liu and Yu, 2017).

Scholars and practitioners are aware about the difficult to forecast the need of users' and market during the NPD process, especially when the time span from idea to launch on the market is stretched (Antioco et al., 2008; Bianchi et al., 2020; Kettunen, 2009; Salvato and Laplume, 2020). As such, the literature on NPD proposed two specular approaches to manage the high degree of uncertainty inherent the NPD process, namely Stage-Gate and Agile (Bianchi et al., 2020). The first attempts to control uncertainty through extensive analysis and planning at the outset of the NPD process. The latter attempts to respond to uncertainty by with a gradual progression of product requirements and plans during the development process. In the middle, Hybrid Agile-Stage-Gate models have been recently developed attempting to gather both the benefits of an upfront and granular project specification and market understanding together with an increased margin of flexibility to adapt the project in due course (Cooper and Sommer, 2016; de Vasconcelos Gomes et al., 2021; Kulk and Verhoef, 2008; Salvato and Laplume, 2020).

On the side of the features management during the NPD process, the literature approached the necessity of continuous adaptation and flexibility by modularity (Burke, 2013; Mikkola, 2003; Yu et al., 2010) while different methods have been explored to accomplish the planning and the selection of features with Real Options approaches (Wouters et al., 2011), technology acceptance models (Burke, 2013), and Big Data driven methods (Tan and Zhan, 2017).

However, the problem of BPs often escape the boundaries set by exiting models and tools to manage the uncertainty of the NPD process, creeping silently through the various phases of the development process as its existence is mostly understudied and underdeveloped, nor appropriately recognised and encompassed in the NPD process management tools (Bianchi et al., 2019, 2020; Bjarnason et al., 2012; Buschmann, 2009, 2010; de Vasconcelos Gomes et al., 2021; Rust et al., 2006; J. B. Schmidt and Calantone, 2002; Thompson et al., 2005)

As a matter of fact, the analysis of the available literature showed that the body of knowledge related to BPs is poorly theoretically developed, with spreading of terminological confusion. In the meantime, the field of NPD has growth and transformed by stressing the importance of novel approaches with fast development cycles, yet not accounting risks associated with BPs (Bianchi et al., 2020; Marzi et al., 2021). As a result, there is a consistent theoretical and empirical gap that needs to be addressed and brought to the attention of scholars, managers, and practitioners.

To address such a gap, the present paper analyses and unpack the BPs' literature by focusing on their causes and their effects, reconnecting them with the latest available studies on the NPD realm. In doing so, an integrative and comprehensive overview capable to clarify the interconnections, the evolution, and the effects of the various forms of BPs is proposed together with a systematization of phenomenon.

The results emerged from the present analysis highlighted the existence of different BPs at different stages of the development process. Furthermore, a new and updated taxonomy of BPs has been developed, clustering with unified terms the exiting terminological mix-up across different fields. BPs have been approached with a clinical-like lens by highlighting the sub-pathologies, the aetiology, and the symptoms associated, considering all of them in a single, general model that shows the interconnection between the possible pathological state of the NPD process resulting from BPs.

In the following paragraph, the methods for the study are presented. Next, the third paragraphs propose the taxonomy and a reclassification of the BPs'. Finally, it is presented the conclusions, the limitations along with an agenda to set the further development of the field. Appendix 1 offers additional methodological notes.

2. Methods

As the main goal of this paper is to offer an integrative review and a comprehensive model to understand BP, the high fragmentation of the literature on the topic required the use of a structured approach to collect, merge, interpret and recompose the puzzle of the available studies on BPs spanning across different fields.

A combination of best practices coming from literature reviews (Cronin and George, 2020; Marzi et al., 2021; Tranfield et al., 2003) and theory development practice (Corley and Gioia, 2011; Wacker, 1998) has been applied. The use of integrative review approach (Cronin and George, 2020) as guiding baseline to analyse, interpret, and expand the extracted material has been used following its ability to summarises the existing state of knowledge on a subject and connects disparate conversations nested in disparate paradigms and fields of study (Cronin and George, 2020).

In doing so, the first step involved a wide research on the major databases (Scopus, Web of Science, EBSCO) using a set of different research terms related to BPs, grounded on the approach of Shmueli et al. (2017).

The research on the databases has been performed iteratively during the development of the present study to ensure the inclusion of the most up-to-date literature available. Queries have been made the 2nd of March 2020; 1st of October 2020; 23rd of April 2021.

Next, the results from the various sources have been collated and compared (Cronin and George, 2020; Marzi et al., 2021; Tranfield et al., 2003).

Only studies referring to excessive development on NPD have been included by combining the definitions of the various facets of BPs arose from previous studies, not only limited to the software domain (Bianchi et al., 2019; Bjarnason et al., 2012; Buschmann, 2009, 2010; Coman and Ronen, 2010; Elliott, 2007; Gregori and Marcone, 2019; Shmueli and Ronen, 2017; Thompson et al., 2005). Though the first step, 39 studies directly relating to topic of BPs have been identified.

Due to the sparse contribution exiting in the BPs field, after reading all the material coming from the first step, a bi-directional analysis of the citations has been performed by looking at all the references cited and citing by the first batch of 39 studies in order to catch possible missing material not emerging from the query (Cronin and George, 2020). Also grey literature including practitioners' contributions, editorials, and comms directly contributing to the topic of BPs has been included. This step allowed the inclusion of 17 additional studies for a final count of 56 documents[2].

3. A Taxonomy of Beyond Pathologies

The term "Beyond Pathologies" defined in the present study is inspired from the works of Ronen and Pass (2008) and Coman and Ronen (2010) on Overspecification and Overdesign wich highlighted the developers' bias of proposing products "beyond" what is needed by the users generating a possible pathological state of the NPD process.

However, the research interest about BPs started much earlier when Paul Abrahams, president of the ACM society, warned about the risk of trying to anticipate and account all the possible future software extension by Overspecification (1988). At the same time, Bohem and Papaccio (1988) pose the same question in term of costs of Overspecification and the limited benefits emerging from it.

A broader interest in BPs materialised some years later when scholars identified the development of highly featured products as a response to the increasing competition of technology-driven industries (B. Boehm et al., 2000; Christensen and Bower, 1996; Davis and Venkatesh, 2004; R. Schmidt et al., 2001). Later, the uncontrolled extension of projects' scope captured the attention of scholar and practitioners highlighting the risk of exposing the entire NPD of failure, low-usability, and unnecessary complexity (B. Boehm and Turner,

[2] As this study is single name authored, in order to double check the validity and reliability of the retrieved data, in line with the best practice on the field (Cronin and George, 2020; Tranfield et al., 2003), it has been asked to another independent scholar to doble-check and confirm the accuracy of the data included in the present study. The response has been positive and confirmed the reliability of the data extraction.

2005; Chen et al., 2009; Dean Hendrix and Schneider, 2002; Knight and Robinson Fayek, 2002; Ropponen and Lyytinen, 2000).

In 2005, a crucial step in defining the outcome of excessive feature-rich has been made by Thompson et al. (2005) who defined Feature Fatigue, while a subsequent theoretical improvement and user centric empirical exploration has been made by different scholars in the late 00s (Buschmann, 2009, 2010; Choi and Bae, 2009; Coman and Ronen, 2010; Gil and Tether, 2011; Gill, 2008; Han et al., 2009).

Grounded on the previous developments, from 2010 BPs received a further theoretical and empirical development reconnecting through the role of uncertainty and decision biases in the generation of them, mostly from the field of software development (Belvedere et al., 2013; Bjarnason et al., 2012; Shmueli et al., 2015, 2016; Shmueli and Ronen, 2017).

More recently, a set of additional experimental studies aimed to empirically asses the magnitude and the BPs' effects has been published since 2019, allowing a more complete understanding and characterisation of phenomenon (Bianchi et al., 2019; De Giovanni, 2019; Garcia et al., 2019; Gregori and Marcone, 2019; Jain, 2019). Also, literature explored the role of features in generating overcomplexity to users and actors involved in the NPD process (Cesaretto et al., 2021; Delpechitre et al., 2019; Eytam et al., 2017, 2020; Verkijika, 2021)

However, the absence of a generalised theoretical framework and the multidisciplinary nature of BPs, originated a proliferation of terms with similar meanings. Thus, the first requirement for a comprehensive and unambiguous development of BPs starts with a terminological and conceptual reclassification of them across all the various industries, from product to software development.

In doing so, Table 1 revises and expands the classification proposed by Shmueli and Ronen (2017) for software development thus proposing a reclassification of the terms used by both scholars and practitioners for dealing BPs in the entire field of NPD. Table 1 regroups synonyms of terms used in the literature, suggesting additional sub-categories, and further detailing the characteristics for each term. BPs are reorganised under the three major categories of “Beyond Needs excesses” (BNE), “Beyond Plans excesses” (BPE), and “Beyond Resources excesses” (BRE).

TABLE 1 ABOUT HERE

3.1 Exploring Beyond Needs Excess (BNE)

Beyond Needs Excess (BNE), refers to specifying, designing, and developing products and services beyond the current needs of the customers or the market. A product or service suffering of beyond needs excess is loaded with superfluous features offering a number of functionalities and/or better performance levels than needed (Bianchi et al., 2019; Coman and Ronen, 2010).

Inside BNE category, Overspecification usually occurs during the initial stages of concept and specifications definition, when additional features, not strictly necessary technologies, and possible future usages of the product or service are predicted and included in the project’s blueprint (Allen et al., 2019; Coman and Ronen, 2010; Wouters et al., 2011; Yu et al., 2010). Overspecification serves also as a padding, a buffer, against the uncertainty of the NPD process by including unnecessary characteristics to anticipate possible future trajectories or future market trends (Allen et al., 2019; Bianchi et al., 2019; Coman and Ronen, 2010; Shmueli et al., 2015). Also, during the specification process, marketing departments usually push to include nice-to-have features aimed to alluring customers and increase the market appeal of the product or service (Bianchi et al., 2019; Hamilton et al., 2017; Rust et al., 2006; Thompson et al., 2005).

Overdesign is similar to Overspecification and share most of its characteristics but occurs later, when features that were not included in the frozen plan of specifications are added during the development phase. Like to Overspecification, Overdesign can be used to create a buffer to avoid later re-design of the product or service (Allen et al., 2019). In such a case, numerous features are partially designed and implemented and later removed in future upgrades of the product of service. Allen (2019) showed that in high complex manufacturing environment, Overdesign can be a complement to modularity and an alternative to re-redesign if the number of features are meticulously controlled and evaluated.

Finally, the Overspecification/Overdesign issue aiming to include fancy features is occasionally named as “bells-and-whistles” (Ropponen and Lyytinen, 2000), while including alluring features is sometimes named as Gold-Plating (B. W. Boehm and Papaccio, 1988)

3.2 Exploring Beyond Plans Excess (BPE)

Beyond Plans Excess (BPE), refers to the tendency to add features by continuously deviate from the plans, specifications, and scope during the project’s execution phase. BPE differs from BNE as it mostly manifests when the features are already planned and specifications are “frozen” as well (Bianchi et al., 2019; Elliott, 2007).

Inside BPE category, Feature Creep and Scope Creep are the most common pathologies. Despite some authors considered in the past these two terms synonyms (Chen et al., 2009; Elliott, 2007), Feature Creep and Scope Creep involve two different aspects of excessive development (Bianchi et al., 2019; Buschmann, 2009; Chen et al., 2009). The first is limited to the features, when several of them are added or changed during the development process. Feature Creep diverges the project from the “optimal” set of features planned to reach the scope of the project.

Scope Creep, instead, involves the entire project with a constant and uncontrolled extension and revision of the scope of the project itself (Buschmann, 2009).

There is a connection between Scope Creep and Feature Creep (Buschmann, 2009). When the scope of a project steadily increases and changes because of Scope Creep, more and different features are requested to address the new scope favouring Feature Creep as a cascade effect (Davies et al., 2016; Gil et al., 2006; Gil and Tether, 2011).

Finally, Featuritis manifests with an escalation of demanding of additional features to cover a broader set of users. The escalation happens in a short time during the development process, especially in the later stages of development (Buschmann, 2010; Eliëns et al., 2018; Repenning, 2001; J. B. Schmidt and Calantone, 2002). Featuritis inevitably affects the features' value, more and more features at the expense of quality (Buschmann, 2010).

3.3 Exploring Beyond Resources Excess (BRE)

Beyond Resources Excess (BRE), explicitly refers to BPs involving the scope of the project by setting, before the start of the development process, the scope of a NPD project beyond the limits of project or company resources (Bjarnason et al., 2012; Shmueli et al., 2015).

BRE only includes Overscoping that mostly originates in fast-moving and market-driven environment when unfocused project goals, lack of communication, and lack developers' involvement during the early phases of project push the entire NPD project to an excessive extension of scope beyond the resources allocated for such project (Bjarnason et al., 2012).

Overscoping typically begins with unrealistic expectations about the project, mostly related to budgets and schedules. Sales staff frequently propose to deliver unrealistically features without considering project management implications, aiming to catch new customers (Bjarnason et al., 2012). Irrational behaviours, together with poor project management practices, promote the definition of a bloated project's scope in relation to the resource available for the project (J. B. Schmidt and Calantone, 2002).

Overscoping could also happen during the development phase if the entire scope of the project is revised taking on the similar characteristics of Scope Creep (Bjarnason et al., 2012). However, while Overscoping and Scope Creep could appear the same pathology, Bjarnason et al. (2012, p. 1108), uses the expression "biting off more that you can chew" which provides the best idea of what Overscoping is. Scope Creep is a steady and slow increase of the scope during the project, Overscoping is an abrupt increase of the scope at the beginning or during the development, if the scope is substantially revised.

4. The physiology of Beyond Pathologies

The previous studies on BPs explored them as separate and independent phenomena. However, by delving into the literature, it emerges that they are rather highly interconnected, reinforcing each other and causing a cascade of effects throughout the entire cycle of the NPD process (Bjarnason et al., 2012; Buschmann, 2009, 2010). In the next sections, BPs will be analysed as a comprehensive and interconnected pathology of the NPD process, summarising and unpacking the findings of the literature that allow to rebuild the evolution of BPs, the antecedents, and symptoms.

Looking at the way they manifest, BPs present structures that typically characterise human diseases (Bianchi et al., 2019; Coman and Ronen, 2010). In particular, BPs manifest themselves as a clinical-like condition with their antecedents (Aetiology) and effects (Symptoms). Consequently, the analysis and the integration of the literature allowed the construction of a comprehensive guiding framework for BPs (Corley and Gioia, 2011; Cronin and George, 2020; Tranfield et al., 2003; Wacker, 1998).

Figure 1 presents how BPs are generally interconnected as well as some of their core features, including their typical occurrence throughout the main phases of the NPD process, namely, scoping, specification, development, and launch phase. BPs and their antecedents tend to concentrate and occur multiple times between the scoping and development phases. For example, an excessive inflow of requirements could happen during the specification and/or the development phase, and this can generate Overspecification and/or Feature Creep (B. Boehm et al., 2000).

FIGURE 1 ABOUT HERE

What it has been label the “symptoms,” instead, tend to manifest towards the late development phase or after the launch. Examples are budget overruns and project delays that can depend on an excessive number of features implemented into a product during its development, which, in turn, can cause Feature Fatigue and quality issues that can only be detected once the product is already on the market.

As shown by Figure 1, the occurrence of BPs at different stages of the NPD process can be due to various reasons. In most cases, this relates to the uncertainty that naturally characterises the NPD process, which forces the inclusion of several margins of tolerance, allowing the readjustment of a product during its whole development (Allen et al., 2019; Antioco et al., 2008; Bianchi et al., 2020; B. Boehm and Turner, 2005; B. W. Boehm and Papaccio, 1988; Kulk and Verhoef, 2008; Long et al., 2021).

The emergence of one or multiple BPs inside the NPD project can be due to a combination of several factors, outlined in Table 2. As mentioned above, this can happen at different stages of the NPD process. For example, when the scope of a NPD project is over expanded beyond the resources of the company and not accurately revisited based on the resources constraints, it is easy to incur in Overscoping (Bjarnason et al., 2012). Likewise, if project managers do not freeze the escalation of inclusion additional features during the development process, a series of adverse pathologies can be generated (Allen et al., 2019; Bianchi et al., 2019; Long et al., 2021; Repenning, 2001).

FIGURE 2 ABOUT HERE

Figure 2 makes a comparison between two alternative situations. Situation A (solid line) shows a NPD project starting from the safe area, creeping into the tolerance area, and back to the safe one, reaching no critical situation associated to BPs. Here are taking place compensation mechanisms such as project management best practices, review of requirements, and scope reduction, NPV value estimation of features (Allen et al., 2019; Bjarnason et al., 2012; Coman and Ronen, 2010; Kulk and Verhoef, 2008; Salvato and Laplume, 2020; Tan and Zhan, 2017). In Situation B (dashed line), the project starts from a safe area but escalate from the margin of tolerance towards pathology. When inside the margin of tolerance, an escalation of adverse conditions or the failure in recognising the early stages of BPs inevitably brings the project to a point of no return where the pathology spread (Alahyari et al., 2019; Brahma and Wynn, 2020; Kulk and Verhoef, 2008; Long et al., 2021; Repenning, 2001; J. B. Schmidt and Calantone, 2002).

As it emerges from the literature, BPs are not necessarily independent from each other but are rather highly interconnected (see Figure 1). In this sense, the boundaries between one pathology and another can be blurred as pathologies are intertwined and tend to reinforce each other (Allen et al., 2019; Bianchi et al., 2019; Shmueli and Ronen, 2017). Moreover, different pathologies can coexist in the same project and one pathology can generate another (Bianchi et al., 2019; Coman and Ronen, 2010). As shown in Figure 1, upstream and scope pathologies (Overscoping, Overspecification, and Scope Creep) may influence or even generate downstream pathologies like Feature Creep, Featuritis, and Overdesign. For example, if the scope of the project is expanded during the development, it generates Scope Creep. In order to address the new, enlarged scope of the NPD project, new features are needed generating Feature Creep (Buschmann, 2009, 2010; De Giovanni, 2019; Elliott, 2007).

To summarise, BPs configure like clinical diseases of the NPD project with a variety of antecedents (Aetiology) and effects (Symptoms). Mild forms of BPs exist in every NPD project as a result of the natural uncertainty inherent the development process itself, but a NPD process can only reach a pathological state when it gets beyond the margin of tolerance which can happen at various stages of the NPD process. Finally, BPs are highly interconnected and tend to create a vicious circle that reinforces their negative effects.

4.1 Antecedents (Aetiology) of Beyond Pathologies

As for clinical conditions, several antecedents can cause and concur to the origin of BPs. Table 2 presents the antecedents identified and examined by the existing literature on NPD, which are here categorised in accordance with their primary drivers. The sign “X” marks the antecedents that, given our current knowledge from the available literature, are likely to have a stronger pathogenic influence on a NPD process. The sign “p” marks the antecedents that possibly contribute to the development of a pathology, although their effects are still unclear or yet to be fully assessed.

As mentioned above, several antecedents could be the source of BPs. More specifically, the relationship between BPs and antecedents is many-to-many: a single antecedent could prompt the development of various BPs, and multiple antecedents can be involved in the development of more pathologies (Elliott, 2007; Shmueli and Ronen, 2017).

The antecedents of BPs can be classified into four main driving areas, namely, uncertainty, market, cognitive, and project driven.

TABLE 2 ABOUT HERE

The first category, uncertainty-driven, includes conditions due to the extreme ambiguity of the external environment, which pushes the NPD project outside its boundaries (Allen et al., 2019; Antioco et al., 2008; Backman et al., 2007; Bianchi et al., 2019; Brahma and Wynn, 2020; Kulk and Verhoef, 2008; Wouters et al., 2011; Yu et al., 2010). In cases of this sort, expanding the scope and the specifications as well as adding extra features represent potential strategies to reduce the fuzziness of the customers' needs and thus the risk of pathology (Bianchi et al., 2019). However, intentionally excessive safe margins can cause a BP, too (Allen et al., 2019; Coman and Ronen, 2010; Kulk and Verhoef, 2008). A poor understanding of the users' needs can lead engineers, developers, R&D Managers, and project managers to include buffers aimed at facilitating possible future upgrades of a product (Brahma and Wynn, 2020; Coman and Ronen, 2010; Wouters et al., 2011). When the technological pace is high (e.g., smartphone industry), or the regulatory environment is turbulent (e.g., telecommunication and cybersecurity industries), including a safe margin during the NPD project helps in anticipating possible and unexpended changes in the needs of the market (Allen et al., 2019; Bjarnason et al., 2012; Brahma and Wynn, 2020; Wouters et al., 2011).

From the side of market-driven antecedents, studies show that marketing and technical departments are often competing in finding the ideal balance between a product's attractiveness to the customers and its technical feasibility (Goodman and Irmak, 2013; Rust et al., 2006; Stock, 2011; Thompson et al., 2005; Thompson and Norton, 2011). Two common antecedents of BPs are the what we can call “leave-all-options open” approach and “one-size-fits-all” approach. The first is an intentional delay in the closure of a project, with the aim of addressing users' needs that can eventually emerge (Backman et al., 2007; Bianchi et al., 2019; Coman and Ronen, 2010; Wouters et al., 2011). For example, Philips developed products with silent and uncomplete features to leave all the options open for future market needs (Wouters et al., 2011). The "one-size-fits-all" approach, instead, refers to the mistaken belief that users prefer an all-rounder – convergent – product that addresses several users' needs with a "swiss knife" approach (Gill, 2008; Goodman and Irmak, 2013; Han et al., 2009). The "one-size-fits-all" approach could also depend on the aim of meeting the demands of both basic and advanced users (Belvedere et al., 2013; Coman and Ronen, 2010; Gill, 2008; Goodman and Irmak, 2013; Jain, 2019).

Market-driven BPs can also depend on the attempt of attracting new or returning customers by asking for unrealistic features or performance (Bjarnason et al., 2012; Eytam et al., 2017, 2020; Rust et al., 2006; Thompson et al., 2005). Thomson and Norton (2011) examined the social aspects of selecting a product with a high number of features and capabilities. The result showed that product with a high number of features and capabilities increase the positive social impression for public display. The users accept a decrease in usability coming from an excessive number of features in return for social impression. This is particularly true for conspicuous consumption (e.g., luxury goods) where consumers prefer feature-rich products for public display. On the contrary, consumers select least featured products when looking for performance and usability, even for public performance display (Thompson and Norton, 2011).

Likewise, including additional features resulting from an excessive acquiescence to users' desires could bring the project to include features of low quality or features with a limited userbase (Bleda et al., 2021; Buschmann, 2010; Shmueli et al., 2015). Finally, competitive pressure plays a crucial role in generating BPs, particularly in mass markets like consumer electronics, domestic appliances, or automotive segments (Rust et al., 2006). Here, features represent competitive leverage to attract new customers, increase the re-purchases, or retain customers (Bleda et al., 2021; Thompson et al., 2005). This results in extreme, potentially pathogenic pressures on marketing and development departments from top management to include new and additional features (Bleda et al., 2021; Christensen and Bower, 1996; Coman and Ronen, 2010).

Moving to cognitive drivers, several studies showed that a variety of cognitive and emotional variables can contribute to BPs, including cognitive biases, emotions, and the general behaviour of project managers, engineers, developers, and R&D Managers (Antioco et al., 2008; Belvedere et al., 2013; Bianchi et al., 2019; Shmueli et al., 2015, 2016). In contexts of uncertainty, human biases such as overconfidence, anchoring, planning fallacy, sunk-cost fallacy, and perfectionism all affect decision making and thus play a crucial role in the entire NPD cycle. For instance, previous research shows that biases in decision-making favour the uncontrolled expansion of product features as the developers tend to excessively overestimate the usefulness of their own developed features (Belvedere et al., 2013; Shmueli et al., 2015). Likewise, the irrational attachment to a project, to a specific technology, or a particular set of features creates a ripple effect that leads to BPs (Bianchi et al., 2019; Shmueli et al., 2015). Several studies showed that developers are affected by a series of cognitive biases during the development process, such as IKEA effect, I-designed-my-self effect or emotional attachment (Bianchi et al., 2019; Franke et al., 2009; Goodman and Irmak, 2013; Norton et al., 2012). Such biases create an overestimation of the value of what has been developed, creating a fertile background for Overdesign. The literature also shows that cognitive styles (e.g., rational vs. intuitive), can be associated to various BPs (Bianchi et al., 2019). Also, developers are highly skilled professionals and “power users” with an attitude to request the state-of-the-art technology that is useful only for a few users (Belvedere et al., 2013; Coman and Ronen, 2010). The dissonance exiting between the perception of value from the developers and the perception of value from the average final users creates the antecedents for proposing products beyond the needs of the market (Bianchi et al., 2019).

Regarding the project-driven antecedents, the quest for continuous improvements can happen to be a double-edged sword for NPD. Project managers, developers, and the other actors directly involved in the NPD process sometimes push to achieve the highest technological level possible (Bianchi et al., 2020; Christensen and Bower, 1996; Davis and Venkatesh, 2004; R. Schmidt et al., 2001), which might involve the implementation of extremely recent technologies (Bianchi et al., 2019; Coman and Ronen, 2010). Such an endless desire of improvement can be dangerous since it can require changes in the product's basic architecture, which paves the way for a variety of issues (Backman et al., 2007; Coman and Ronen, 2010). The optimal trade-off between the advantages and disadvantages of a new but yet untested technology should be considered, especially during the later developmental stages (Christensen and Bower, 1996; Davis and Venkatesh, 2004; R. Schmidt et al., 2001). Similarly, a continuous and uncontrolled stream of requirement inflow during both development and specification phase is a potential cause of BPs (Bjarnason et al., 2012). In this specific case, if project managers do not mediate between the requests of the market and the development teams by reducing the requirement inflows, the risk of BPs increases proportionally with the variety of new requirements (Bjarnason et al., 2012).

As regards the resources and goals of the project, the poor assessment of the financial and time resources available for a project generates a tendency to overshoot the size and scope of the project (Bjarnason et al., 2012; Chen et al., 2009). Equally, unfocused project goals are antecedents of scope and feature bloat due to the fuzziness of the project's boundaries (Bjarnason et al., 2012; Jain, 2019). Finally, the low involvement of the development team during the specification phase can create a misalignment between the technical feasibility of specific features of a product and the customers' needs (Coman and Ronen, 2010). Technical teams must be involved in the specification phase to avoid the inclusion of unrealistic or unfeasible features proposed by marketing or selling teams (Bjarnason et al., 2012; Coman and Ronen, 2010).

As for clinical diseases, the aetiology of BPs is complex and multifactorial, involving single antecedents of major effects as well as multiple factors concurring altogether in generating a given BP. Indeed, the antecedents of BPs depend on uncertainty from a variety of sources, ranging from external factors (e.g., shifts of customers' preferences, competitors' strategic moves, nascent technological trajectories) to individual-level factors (developers' creative responses to discoveries during the project, cognitive biases, the co-evolution of technical solutions in the interacting product components).

4.2 Symptoms of Beyond Pathologies

As anticipated in Figure 1, BPs are characterised by a series of tangible symptoms. Table 3 presents a list of such symptoms grouped in two categories: first, symptoms emerging during the NPD process and, second, those emerging after the launch of a product. As for Table 2, the sign "X" marks symptoms that are well-recognised within the literature and are very likely to occur in the context of a given BP. The sign "p" marks the symptoms that can possibly occur, although their effects are still unclear or not fully explored.

TABLE 3 ABOUT HERE

As shown in Table 3, the two most widespread symptoms of BPs are Feature Fatigue and quality issues. Feature Fatigue, explored by Thompson et al. (2005), is the difficulty experienced by the users with products that offer an extensive and excessive set of features and capabilities (Rust et al., 2006; Thompson et al., 2005). When the number of feature is excessive, they create an overcomplexity for users resulting in a fatigue during the regular usage of the product or service (Rust et al., 2006; Thompson et al., 2005).

As Thompson et al. (2005) show, three desirable outcomes can depend on the number of features of a product (Figure 3).

The first desirable outcome, represented in Figure 3 as Point A, maximises the net value for customers by offering a product with a proportionate set of features. The second possible outcome, Point B, exploits initial purchase by offering an alluring but highly featured product. The latter, Point C, maximises repurchase by offering a simple but reliable product with a limited number of features. As a result, depending on the organisations' goal, the boundaries of the product should be set between points B and C (Thompson et al., 2005). Figure 3 highlights the effects of an excessive features selection - above point B the product has an excessive number of features resulting in Feature Fatigue for the users (Thompson et al., 2005).

FIGURE 3 ABOUT HERE

For example, an Overscoped project would result in an excessively featured delivered product or a "one-size-fits-all" approach could result in an Overspecified and/or Overdesigned product with several extra features that move the product beyond the point B on Figure 3, thus resulting on Feature Fatigue (Stock, 2011). A similar situation happens when the product is loaded with features aimed to allure and attract customers result in additional resources needed for development and less usability for users. Such evidences are confirmed by the later studies of Bjarnason (2012), Elliot (2007), Buschmann (2009, 2010), Stock (2011), Delpechitre (2019), and Jain (2019). The current body of knowledge about BPs, the literature often suggests indirectly that Feature Fatigue is one of the final, most tangible outcomes of

BPs (Bjarnason et al., 2012; Coman and Ronen, 2010; Elliott, 2007). Paradoxically, such a manifest relation is not sufficiently empirically explored, and Feature Fatigue is rarely reconnected with the drawbacks happing during the NPD process (Coman and Ronen, 2010; Jain, 2019).

Moving to quality issues and consumers' expectation not met, the literature showed a significant correlation between BPs and the aforementioned tangible outcomes (Bjarnason et al., 2012; Coman and Ronen, 2010; Elliott, 2007; Gregori and Marcone, 2019; Shmueli et al., 2016). When extra features are added, each of them competes and clash for the resources allocated to the project in term of financial resources, time, and attention devoted to quality (Bianchi et al., 2019; Bjarnason et al., 2012). Increasing workloads due to endless requirements and features inflow defocus the attention and the resources devoted on each feature. Same resources, more feature, less attention allocated to each feature (Bjarnason et al., 2012). The typical result is a product that is delivered with some low-quality features, which results in various sorts of malfunctioning that cannot meet the expectation of the final users, see the example of Mercedes cited in the introduction (Bianchi et al., 2019; Bjarnason et al., 2012; Davis and Venkatesh, 2004). Bloat in features or scope does not influence only Feature Fatigue but also affect the quality of the whole NPD project in a rapid escalation of adverse, cumulative effects (Buschmann, 2010; Coman and Ronen, 2010; Davis and Venkatesh, 2004; Gill, 2008; J. B. Schmidt and Calantone, 2002). Quality issues are frequently interrelated with Feature Fatigue and the two often goes hand by hand (Rahman and Manzur Rahman, 2009; Rust et al., 2006; Thompson et al., 2005).

BPs symptoms are not limited to the launch and after-launch phases and might also happen during the NPD stage that comes before the launch. The symptoms emerging during the launch and after-launch phases are connected with the previous NPD stages of development and they include budget overruns, project delays, and loss of focus (Coman and Ronen, 2010;

Garcia et al., 2019; Gil and Tether, 2011; J. B. Schmidt and Calantone, 2002; Shmueli and Ronen, 2017). The underlying human predisposition to consider every effort linear does not account that each additional feature added to the project exponentially increase the complexity of the systems' architecture (Coman and Ronen, 2010; Garcia et al., 2019; Shabi et al., 2021). Many BPs symptoms are due to an underestimation of how additional features will impact the NPD process, such as the exponential (rather than linear) increasing of the complexity of a systems' architecture (Alahyari et al., 2019; Backman et al., 2007; Coman and Ronen, 2010).

Additional features compete in assimilating the resources allocated to the project; more features implies less time devoted to each feature, plus additional costs for the whole project (Alahyari et al., 2019; Bjarnason et al., 2012; Buschmann, 2010; Davies et al., 2016; Gil et al., 2006; Gil and Tether, 2011; Thal et al., 2010). The enthusiasm coming with a highly capable and feature-rich product inevitably clashes with the significant refactoring needed to complete, integrate, and test an extensive set of features that will likely generate massive costs and schedule slips (Buschmann, 2010; Eliëns et al., 2018). The clash between competing features results in budget overruns, delays, and defocused scope (Bjarnason et al., 2012; Coman and Ronen, 2010; Elliott, 2007; Shmueli et al., 2015).

5. Conclusions, Open questions, and Limitations

The present study defines the phenomenon of BPs and proposed an integrative framework to clarify such a widespread but under-explored phenomenon. Grounded on the theoretical and empirical findings on the topic, the present research takes the stock of situation on BPs serving as a springboard for further advancements on the topic.

However, the multifaced topic of BPs still poses several questions to scholars and practitioners that needs further empirical and conceptual investigation. The three key aspects of BPs yet deserve to be better explored, namely, the antecedents, the symptoms, and the nature of BPs themselves.

Antecedents appear to be the most studied aspects of BPs so far. Although the complete aetiology of BPs is yet to be clarified, previous studies identified the involvement of cognitive biases (Belvedere et al., 2013; Bianchi et al., 2019; Shmueli et al., 2015, 2016) and dysfunctional project management practices (Bjarnason et al., 2012). However, there is still a need to identify additional specific factors that can trigger BPs and take the appropriate counteractions. For example, it may turn out to be essential to rebuild the entire decisional chain that leads to the emergence of BPs, both at the scoping and development phase. While some internal antecedents have been explored (Bjarnason et al., 2012; Coman and Ronen, 2010; Garcia et al., 2019), it is necessary to advance our understanding of how, and to what extent, external and unpredictable variables, such as market turmoil and technological pace, can affect the health of a NPD project.

Moving to the symptoms of BPs, the current state of knowledge is highly fragmented. While some studies explored the effects of BPs on product and project performance (Belvedere et al., 2013; Bianchi et al., 2019; Bjarnason et al., 2012; Chen et al., 2009; Choi and Bae, 2009; De Giovanni, 2019; Garcia et al., 2019; Gregori and Marcone, 2019), there is a need to understand the impact of BPs on specific aspects of product and project performance such as costs, speed, quality, customer satisfaction, and product placement. Previous studies provide us with the basis for studying the negative effects of BPs on NPD, but is unclear at what stages of the process, how, and to what extend BPs affect the NPD performance (Bjarnason et al., 2012). Other crucial aspects of BP symptoms relate to the role of Feature Fatigue, which appear to be strongly connected with BPs (De Giovanni, 2019; Rust et al., 2006; Thompson et al., 2005; Wu et al., 2015). As Thompson et al. (2005) showed, the role of features in product success and placement is crucial even in term of the company's product portfolio and overall strategy. The present study highlighted a critical gap in the literature and a severe underestimation of the role of BPs in Feature Fatigue literature. A possible solution lies on including additional scalability and modularity to product and services, allowing an ad-hoc customisation and segmentation starting from a basic platform (Kettunen, 2009; Mikkola, 2003; Wouters et al., 2011).

A final key open question relates to the absence of a multidimensional tool capable of measuring and assessing at least the main BPs. In this sense, seminal studies proposed tools to measure certain, but limited, facets of BPs. Bianchi et al. (2019) proposes a scale to measure sub-constructs underlying BNE, while Belvedere et al. (2013) specifically focus on Overdesign. Other studies approached BPs through experiments (Shmueli et al., 2015, 2016) or qualitative/case analyses (Bjarnason et al., 2012; Garcia et al., 2019; Jain, 2019; Shabi et al., 2021; Wouters et al., 2011). However, a comprehensive measurement tool for BPs is missing. It is of utmost importance to develop tools that can identify specific the type of BP at stake in any given phase of NPD, from scoping to post-launch review, and allow the resolution of the pathology that might emerge (B. Boehm and Turner, 2005). This will prompt the next phase of BPs studies, namely, developing practical decision-making tools and conceptual frameworks for NPD.

Regarding the limitations of the present study, in summarizing the findings, the reflections and the suggestions of the researches, the richness and the depth of each single study focusing only on a precise piece of BPs is inevitably lost. Also, as the present paper is grounded on the literature already available on the topic, it has been possible to propose a theoretical advancement only until the existing evidences allowed.

It therefore clearly emerges the urgent need to have additional empirical study to better understand the phenomenon of BPs. There are still many obscure aspects to investigate, for example, the interconnection among BPs and the portfolio strategy of the company, the role of the NPD practices in mitigating the BPs, and the conscious or unconscious human behaviour that favouring the emergence of such pathologies. Nonetheless, the growing available studies in the last few years forebode a promising future for such a field of research.

References

Abrahams, P. (1988). President’s letter - Specifications and Illusions. *Communications of the ACM*, *31*(5), 480–481.

Alahyari, H., Gorschek, T., and Berntsson Svensson, R. (2019). An exploratory study of waste in software development organizations using agile or lean approaches: A multiple case study at 14 organizations. *Information and Software Technology*, *105*, 78–94. https://doi.org/10.1016/j.infsof.2018.08.006

Allen, J. D., Stevenson, P. D., Mattson, C. A., and Hatch, N. W. (2019). Over-Design Versus Redesign as a Response to Future Requirements. *Journal of Mechanical Design, Transactions of the ASME*, *141*(3). https://doi.org/10.1115/1.4042335

Antioco, M., Moenaert, R. K., and Lindgreen, A. (2008). Reducing ongoing product design decision-making bias. *Journal of Product Innovation Management*, *25*(6), 528–545. https://doi.org/10.1111/j.1540-5885.2008.00320.x

Backman, M., Börjesson, S., and Setterberg, S. (2007). Working with concepts in the fuzzy front end: Exploring the context for innovation for different types of concepts at Volvo Cars. *R and D Management*, *37*(1), 17–28. https://doi.org/10.1111/j.1467-9310.2007.00455.x

Belvedere, V., Grando, A., and Ronen, B. (2013). Cognitive biases, heuristics, and overdesign: An investigation on the unconscious mistakes of industrial designers and on their effects on product offering. In *Behavioral Issues in Operations Management: New Trends in Design, Management, and Methodologies* (Vol. 9781447148, pp. 125–139).

Springer. https://doi.org/10.1007/978-1-4471-4878-4_6

Bianchi, M., Marzi, G., and Guerini, M. (2020). Agile, Stage-Gate and their combination: Exploring how they relate to performance in software development. *Journal of Business Research*, *110*, 538–553. https://doi.org/10.1016/j.jbusres.2018.05.003

Bianchi, M., Marzi, G., Zollo, L., and Patrucco, A. (2019). Developing software beyond customer needs and plans: an exploratory study of its forms and individual-level drivers. *International Journal of Production Research*, *57*(22), 7189–7208. https://doi.org/10.1080/00207543.2019.1581953

Bjarnason, E., Wnuk, K., and Regnell, B. (2012). Are you biting off more than you can chew? A case study on causes and effects of overscoping in large-scale software engineering. *Information and Software Technology*, *54*(10), 1107–1124. https://doi.org/10.1016/j.infsof.2012.04.006

Bleda, M., Querbes, A., and Healey, M. (2021). The influence of motivational factors on ongoing product design decisions. *Journal of Business Research*, *129*, 562–569. https://doi.org/10.1016/j.jbusres.2020.02.018

Boehm, B., Port, D., and Al-Said, M. (2000). Avoiding the software model-clash spiderweb. *Computer*, *33*(11), 120–122. https://doi.org/10.1109/2.881698

Boehm, B., and Turner, R. (2005). Management challenges to implementing agile processes in traditional development organizations. *IEEE Software*, *22*(5), 30–39. https://doi.org/10.1109/MS.2005.129

Boehm, B. W., and Papaccio, P. N. (1988). Understanding and Controlling Software Costs. *IEEE Transactions on Software Engineering*, *14*(10), 1462–1477. https://doi.org/10.1109/32.6191

Brahma, A., and Wynn, D. C. (2020). Margin value method for engineering design improvement. *Research in Engineering Design*, *31*(3), 353–381.

https://doi.org/10.1007/s00163-020-00335-8

Burke, P. F. (2013). Seeking simplicity in complexity: The relative value of ease of use (EOU)-based product differentiation. *Journal of Product Innovation Management, 30*(6), 1227–1241. https://doi.org/10.1111/jpim.12056

Buschmann, F. (2009). Learning from failure, Part 1: Scoping and requirements woes. *IEEE Software*, *26*(6), 68–69. https://doi.org/10.1109/MS.2009.179

Buschmann, F. (2010). Learning from failure, Part 2: Featuritis, performitis, and other diseases. *IEEE Software*, *27*(1), 10–11. https://doi.org/10.1109/MS.2010.14

Cesaretto, R., Buratto, A., and De Giovanni, P. (2021). Mitigating the feature fatigue effect for smart products through digital servitization. *Computers and Industrial Engineering*, *156*, 107218. https://doi.org/10.1016/j.cie.2021.107218

Chen, C. C., Law, C. C. H., and Yang, S. C. (2009). Managing ERP implementation failure: A project management perspective. *IEEE Transactions on Engineering Management*, *56*(1), 157–170. https://doi.org/10.1109/TEM.2008.2009802

Choi, K. S., and Bae, D. H. (2009). Dynamic project performance estimation by combining static estimation models with system dynamics. *Information and Software Technology*, *51*(1), 162–172. https://doi.org/10.1016/j.infsof.2008.03.001

Christensen, C. M., and Bower, J. L. (1996). Customer power, strategic investment, and the failure of leading firms. *Strategic Management Journal*, *17*(3), 197–218.

Coman, A., and Ronen, B. (2010). Icarus' predicament: Managing the pathologies of overspecification and overdesign. *International Journal of Project Management*, *28*(3), 237–244. https://doi.org/10.1016/j.ijproman.2009.05.001

Cooper, R. G., and Sommer, A. F. (2016). The Agile–Stage-Gate Hybrid Model: A Promising New Approach and a New Research Opportunity. *Journal of Product Innovation Management*, *33*(5), 513–526. https://doi.org/10.1111/jpim.12314

Corley, K., and Gioia, D. (2011). Building theory about theory building: What constitutes a theoretical contribution? *Academy of Management Review*, *36*(1), 12–32. https://doi.org/10.5465/amr.2009.0486

Cronin, M. A., and George, E. (2020). The Why and How of the Integrative Review. *Organizational Research Methods*, 1094428120935507. https://doi.org/10.1177/1094428120935507

Damian, D., and Chisan, J. (2006). An empirical study of the complex relationships between requirements engineering processes and other processes that lead to payoffs in productivity, quality, and risk management. *IEEE Transactions on Software Engineering*, *32*(7), 433–453. https://doi.org/10.1109/TSE.2006.61

Davies, A., Dodgson, M., and Gann, D. (2016). Dynamic Capabilities in Complex Projects: The Case of London Heathrow Terminal 5. *Project Management Journal*, *47*(2), 26–46. https://doi.org/10.1002/pmj.21574

Davis, F. D., and Venkatesh, V. (2004). Toward preprototype user acceptance testing of new information systems: Implications for software project management. *IEEE Transactions on Engineering Management*, *51*(1), 31–46. https://doi.org/10.1109/TEM.2003.822468

De Giovanni, P. (2019). A feature fatigue supply chain game with cooperative programs and ad-hoc facilitators. *International Journal of Production Research*, *57*(13), 4166–4186. https://doi.org/10.1080/00207543.2018.1519264

de Vasconcelos Gomes, L. A., Seixas Reis de Paula, R. A., Figueiredo Facin, A. L., Chagas Brasil, V., and Sergio Salerno, M. (2021). Design principles of hybrid approaches in new product development: a systematic literature review. *R and D Management*, *In Press*. https://doi.org/10.1111/radm.12476

Dean Hendrix, T., and Schneider, M. P. (2002). NASA's TReK Project: A Case Study in Using the Spiral Model of Software Development. *Communications of the ACM*, *45*(4), 152–159. https://doi.org/10.1145/505248.506004

Delpechitre, D., Black, H. G., and Farrish, J. (2019). The dark side of technology: examining the impact of technology overload on salespeople. *Journal of Business and Industrial Marketing*, *34*(2), 317–337. https://doi.org/10.1108/JBIM-03-2017-0057

Edwards, C. (2010). The killer iPhone. How Apple designed its new model to be sweet to customers an vicious to rivals. *Strategic Direction*, *26*(11), 25–26. https://doi.org/10.1108/sd.2010.05626kad.005

Eliëns, R., Eling, K., Gelper, S., and Langerak, F. (2018). Rational Versus Intuitive Gatekeeping: Escalation of Commitment in the Front End of NPD. *Journal of Product Innovation Management*, *35*(6), 890–907. https://doi.org/10.1111/jpim.12452

Elliott, B. (2007). Anything is possible: Managing feature creep in an innovation rich environment. *2007 IEEE International Engineering Management Conference*, 304–307. https://doi.org/10.1109/IEMC.2007.5235049

Eytam, E., Lowengart, O., and Tractinsky, N. (2020). Effects of visual simplicity in product design and individual differences in preference of interactive products. *Review of Managerial Science*, 1–43. https://doi.org/10.1007/s11846-020-00391-3

Eytam, E., Tractinsky, N., and Lowengart, O. (2017). The paradox of simplicity: Effects of role on the preference and choice of product visual simplicity level. *International Journal of Human Computer Studies*, *105*, 43–55. https://doi.org/10.1016/j.ijhcs.2017.04.001

Franke, N., Schreier, M., and Kaiser, U. (2009). The “I Designed It Myself” Effect in Mass Customization. *Management Science*, *56*(1), 125–140. https://doi.org/10.1287/mnsc.1090.1077

Garcia, J. J., Pettersen, S. S., Rehn, C. F., Erikstad, S. O., Brett, P. O., and Asbjørnslett, B. E. (2019). Overspecified vessel design solutions in multi-stakeholder design problems.

Research in Engineering Design, *30*(4), 473–487. https://doi.org/10.1007/s00163-019-00319-3

Gil, N., and Tether, B. S. (2011). Project risk management and design flexibility: Analysing a case and conditions of complementarity. *Research Policy*, *40*(3), 415–428. https://doi.org/10.1016/j.respol.2010.10.011

Gil, N., Tommelein, I. D., and Schruben, L. W. (2006). External change in large engineering design projects: The role of the client. *IEEE Transactions on Engineering Management*, *53*(3), 426–439. https://doi.org/10.1109/TEM.2006.877447

Gill, T. (2008). Convergent products: What functionalities add more value to the base? *Journal of Marketing*, *72*(2), 46–62. https://doi.org/10.1509/jmkg.72.2.46

Goodman, J. K., and Irmak, C. (2013). Having versus consuming: Failure to estimate usage frequency makes consumers prefer multifeature products. *Journal of Marketing Research*, *50*(1), 44–54. https://doi.org/10.1509/jmr.10.0396

Gregori, G. L., and Marcone, M. R. (2019). R&D and manufacturing activities regarding managerial effectiveness and open strategy: an industry focus on luxury knitwear firms. *International Journal of Production Research*, *57*(18), 5787–5800. https://doi.org/10.1080/00207543.2018.1550271

Hamilton, R. W., Rust, R. T., and Dev, C. S. (2017). Which features increase customer retention? *MIT Sloan Management Review*, *58*(2), 79–84.

Han, J. K., Chung, S. W., and Sohn, Y. S. (2009). Technology convergence: When do consumers prefer converged products to dedicated products? *Journal of Marketing*, *73*(4), 97–108. https://doi.org/10.1509/jmkg.73.4.97

Jain, S. (2019). Time inconsistency and product design: A strategic analysis of feature creep. *Marketing Science*, *38*(5), 835–851. https://doi.org/10.1287/mksc.2019.1170

Kettunen, P. (2009). Adopting key lessons from agile manufacturing to agile software product development-A comparative study. *Technovation, 29*(6–7), 408–422. https://doi.org/10.1016/j.technovation.2008.10.003

Knight, K., and Robinson Fayek, A. (2002). Use of Fuzzy Logic for Predicting Design Cost Overruns on Building Projects. *Journal of Construction Engineering and Management*, *128*(6), 503–512. https://doi.org/10.1061/(asce)0733-9364(2002)128:6(503)

Kulk, G. P., and Verhoef, C. (2008). Quantifying requirements volatility effects. *Science of Computer Programming*, *72*(3), 136–175. https://doi.org/10.1016/j.scico.2008.04.003

Landis, L., Waligora, S., Mcgarry, F., Pajerski, R., Stark, M., Johnson, K. O., and Cover, D. (1992). Recommended Approach to Software Development Revision 3. In *NASA Goddard Space Flight Center - Software Engineering Laboratory Series*. https://ntrs.nasa.gov/citations/19930009672

Liu, N., and Yu, R. (2017). Identifying design feature factors critical to acceptance and usage behavior of smartphones. *Computers in Human Behavior, 70,* 131–142. https://doi.org/10.1016/j.chb.2016.12.073

Long, D., Morkos, B., and Ferguson, S. (2021). Toward Quantifiable Evidence of Excess' Value Using Personal Gaming Desktops. *Journal of Mechanical Design*, *143*(3), 1–34. https://doi.org/10.1115/1.4049520

Marzi, G., Ciampi, F., Dalli, D., and Dabic, M. (2021). New Product Development during the Last Ten Years: The Ongoing Debate and Future Avenues. *IEEE Transactions on Engineering Management*, *68*(1), 330–344. https://doi.org/10.1109/TEM.2020.2997386

Mikkola, J. H. (2003). Modularity, component outsourcing, and inter-firm learning. *R and D Management*, *33*(4), 439–454. https://doi.org/10.1111/1467-9310.00309

Mitchener, J. (2008). Less simplicity. *Engineering and Technology*, *3*(9), 76. https://doi.org/10.1049/et:20080929

Norton, M. I., Mochon, D., and Ariely, D. (2012). The IKEA effect: When labor leads to love.

Journal of Consumer Psychology, *22*(3), 453–460. https://doi.org/10.1016/j.jcps.2011.08.002

Qin, J., and van der Rhee, B. (2021). From trash to treasure: A checklist to identify high-potential NPD projects from previously rejected projects. *Technovation*, 102259. https://doi.org/10.1016/j.technovation.2021.102259

Rahman, M., and Manzur Rahman, M. (2009). To defeat feature fatigue the right way, understand it first. *Strategic Direction*, *25*(6), 26–28. https://doi.org/10.1108/02580540910952190

Ren, H., and Zhao, Y. (2021). Technology opportunity discovery based on constructing, evaluating, and searching knowledge networks. *Technovation*, *101*, 102196. https://doi.org/10.1016/j.technovation.2020.102196

Repenning, N. P. (2001). Understanding fire fighting in new product development. *Journal of Product Innovation Management*, *18*(5), 285–300. https://doi.org/10.1016/S0737-6782(01)00099-6

Ronen, B., and Pass, S. (2008). *Focused operations management: achieving more with existing resources.* John Wiley & Sons.

Ropponen, J., and Lyytinen, K. (2000). Components of software development risk: how to address them? A project manager survey. *IEEE Transactions on Software Engineering*, *26*(2), 98–112. https://doi.org/10.1109/32.841112

Rust, R. T., Thompson, D. V., and Hamilton, R. W. (2006). Defeating feature fatigue. *Harvard Business Review*, *84*(2), 37–47.

Salvato, J. J., and Laplume, A. O. (2020). Agile Stage-Gate Management (ASGM) for physical products. *R and D Management*, *50*(5), 631–647. https://doi.org/10.1111/radm.12426

Schmidt, J. B., and Calantone, R. J. (2002). Escalation of commitment during new product development. *Journal of the Academy of Marketing Science*, *30*(2), 103–118. https://doi.org/10.1177/03079459994362

Schmidt, R., Lyytinen, K., Keil, M., and Cule, P. (2001). Identifying software project risks: An international Delphi study. *Journal of Management Information Systems*, *17*(4), 5–36. https://doi.org/10.1080/07421222.2001.11045662

Shabi, J., Reich, Y., Robinzon, R., and Mirer, T. (2021). A decision support model to manage overspecification in system development projects. *Journal of Engineering Design*, 1–23. https://doi.org/10.1080/09544828.2021.1908970

Shmueli, O., Pliskin, N., and Fink, L. (2015). Explaining over-requirement in software development projects: An experimental investigation of behavioral effects. *International Journal of Project Management*, *33*(2), 380–394. https://doi.org/10.1016/j.ijproman.2014.07.003

Shmueli, O., Pliskin, N., and Fink, L. (2016). Can the outside-view approach improve planning decisions in software development projects? *Information Systems Journal*, *26*(4), 395–418. https://doi.org/10.1111/isj.12091

Shmueli, O., and Ronen, B. (2017). Excessive software development: Practices and penalties. *International Journal of Project Management*, *35*(1), 13–27. https://doi.org/10.1016/j.ijproman.2016.10.002

Stock, R. M. (2011). How does product program innovativeness affect customer satisfaction? A comparison of goods and services. *Journal of the Academy of Marketing Science*, *39*(6), 813–827. https://doi.org/10.1007/s11747-010-0215-4

Tan, K. H., and Zhan, Y. (2017). Improving new product development using big data: a case study of an electronics company. *R and D Management*, *47*(4), 570–582. https://doi.org/10.1111/radm.12242

Thal, A. E., Cook, J. J., and White, E. D. (2010). Estimation of Cost Contingency for Air

Force Construction Projects. *Journal of Construction Engineering and Management*, *136*(11), 1181–1188. https://doi.org/10.1061/(asce)co.1943-7862.0000227

Thompson, D. V., Hamilton, R. W., and Rust, R. T. (2005). Feature fatigue: When product capabilities become too much of a good thing. *Journal of Marketing Research*, *42*(4), 431–442. https://doi.org/10.1509/jmkr.2005.42.4.431

Thompson, D. V., and Norton, M. I. (2011). The social utility of feature creep. *Journal of Marketing Research*, *48*(3), 555–565. https://doi.org/10.1509/jmkr.48.3.555

Tranfield, D., Denyer, D., and Smart, P. (2003). Towards a Methodology for Developing Evidence-Informed Management Knowledge by Means of Systematic Review. *British Journal of Management*, *14*(3), 207–222. https://doi.org/10.1111/1467-8551.00375

Verkijika, S. F. (2021). Download or swipe left: The role of complexity, future-oriented emotions and feature overload. *Telematics and Informatics*, *60*, 101579. https://doi.org/10.1016/j.tele.2021.101579

Wacker, J. G. (1998). A definition of theory: Research guidelines for different theory-building research methods in operations management. *Journal of Operations Management*, *16*(4), 361–385. https://doi.org/10.1016/s0272-6963(98)00019-9

Wouters, M., Workum, M., and Hissel, P. (2011). Assessing the product architecture decision about product features - a real options approach. *R and D Management*, *41*(4), 393–409. https://doi.org/10.1111/j.1467-9310.2011.00652.x

Wu, M., Wang, L., Long, H., and Li, M. (2015). Feature fatigue analysis in product development. *Total Quality Management and Business Excellence*, *26*(1–2), 218–232. https://doi.org/10.1080/14783363.2013.860697

Yu, A. S. O., Figueiredo, P. S., and De Souza Nascimento, P. T. (2010). Development resource planning: Complexity of product development and the capacity to launch new products. *Journal of Product Innovation Management*, *27*(2), 253–266.

Tables

Table 1 - Taxonomy

	Term	Definition	Synonym(s)	Main References
Beyond needs excess (BNE)	Overspecification	Defining product or service specifications beyond the actual needs of the customer or the market (Ronen and Pass, 2008, p. 162)	Over-requirement*; Gold-plating; Bells-and-whistles; Flexibilitis**; Overshooting	(Bianchi et al., 2019; B. W. Boehm and Papaccio, 1988; Buschmann, 2010; Coman and Ronen, 2010; Shmueli et al., 2015)
Beyond needs excess (BNE)	Overdesign	Developing products or services beyond what is required by the specifications and/or the requirements of the customer or the market (Ronen and Pass, 2008, p. 162)	Performitis; Flexibilitis**	(Allen et al., 2019; Buschmann, 2010; Christensen and Bower, 1996; Coman and Ronen, 2010)
Beyond plans excess (BPE)	Feature Creep	Changes in features while a product [or a service – a.n.] is still in development (Elliott, 2007, p. 304)	Feature bloat; Requirements Creep	(Choi and Bae, 2009; Damian and Chisan, 2006; Davis and Venkatesh, 2004; Elliott, 2007; Rust et al., 2006)
Beyond plans excess (BPE)	Featuritis	Tendency to trade functional coverage for quality – the more functions and the earlier they're delivered, the better (Buschmann, 2010, p. 10)	Feature bloat	(Buschmann, 2010; Elliott, 2007; Hamilton et al., 2017; Rust et al., 2006)
Beyond plans excess (BPE)	Scope Creep	Steady increase of the system's [or project's – a.n.] scope (Buschmann, 2009, p. 68)	Mission Creep; Requirements Creep	(Buschmann, 2009; Chen et al., 2009; Choi and Bae, 2009; Gil and Tether, 2011; Knight and Robinson Fayek, 2002; R. Schmidt et al., 2001)
Beyond resources excess (BRE)	Overscoping	Setting a [project's – a.n.] scope that requires more resources than are available (Bjarnason et al., 2012, p. 1107)	Scope overload	(Bjarnason et al., 2012; Shmueli et al., 2016)

*Over-requirement term is mostly used in the software development realm as a synonym of Overspecification, even if there is a difference between Requirements and Specifications. Requirement refers to what the user needs, while Specification refers to how the software fulfils the user needs.
**If the product is overspecified or overdesigned intending to add extra but unneeded architectural flexibility, like "just in case" functionalities.
"a.n." stands for "author note".

Table 2 - Antecedents

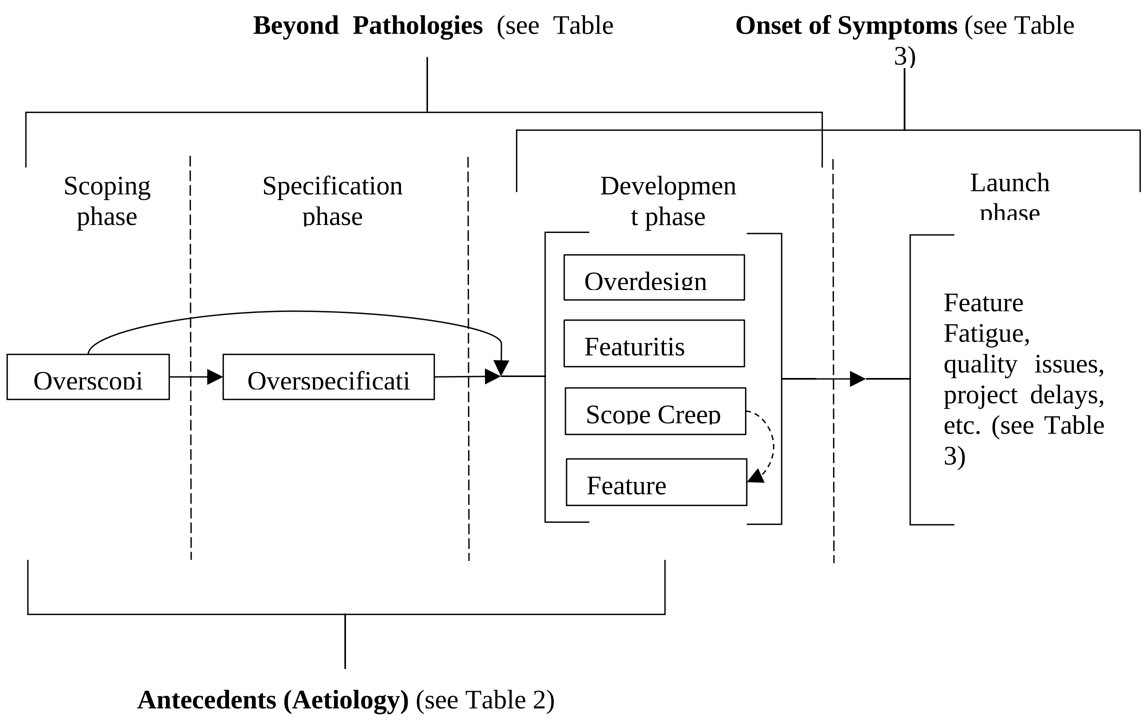
	Antecedents (Aetiology)	**Pathologies**					
		BNE		**BPE**			**BRE**
		Overspecification	*Overdesign*	*Feature Creep*	*Featuritis*	*Scope Creep*	*Overscoping*
Uncertainty driven	Fuzzy front-end	X	X	p		X	p
	Excessive safe margins (buffers) against uncertainty	X	X	p		p	p
	Rapid technological change	X	X	X		p	p
	Changes in legal/regulatory environment during the project		X	X		X	
Market driven	Leave-all-options open approach	X	p	p	p	X	p
	One-size-fits-all approach	X	X	X		X	p
	Attract new customers	X		X	X		X
	Excessive acquiescence to consumers requests	X	p	p	X	p	X
	Competitive pressure	X	X	X	X	X	p
Cognitive driven	Cognitive biases	X	X	p	p	X	X
	Cognitive styles	X	X		p	p	
Project driven	Looking for continuous improvement	X	X	X		p	
	Continuous requirements inflow		X	X	p	p	X
	Unclear overview of the available resources		X			p	X
	Unfocused project goals		p			X	p
	Low involvement of development team/stakeholders during specification phase	X	X			X	X

Table 3 - Outcomes

	Beyond Pathologies	Symptoms					
		After Launch			During Development		
		Feature Fatigue	*Quality issues*	*Costumers' expectations not met*	*Budget overruns*	*Project Delays*	*Project lost focus*
BNE	Overspecification	X	X	p	X	p	X
	Overdesign	X	X	p	X	X	X
BPE	Feature Creep	X	X	p	X	X	p
	Featuritis	X	X	X	p	p	p
	Scope Creep	X	X	p	X	X	X
BRE	Overscoping	X	X	X	X	X	X

Figures

Figure 1 – BPs



Giacomo Marzi, U. of Trieste, Italy, giacomomrz@gmail.com

Figure 2 – Evolution of BPs

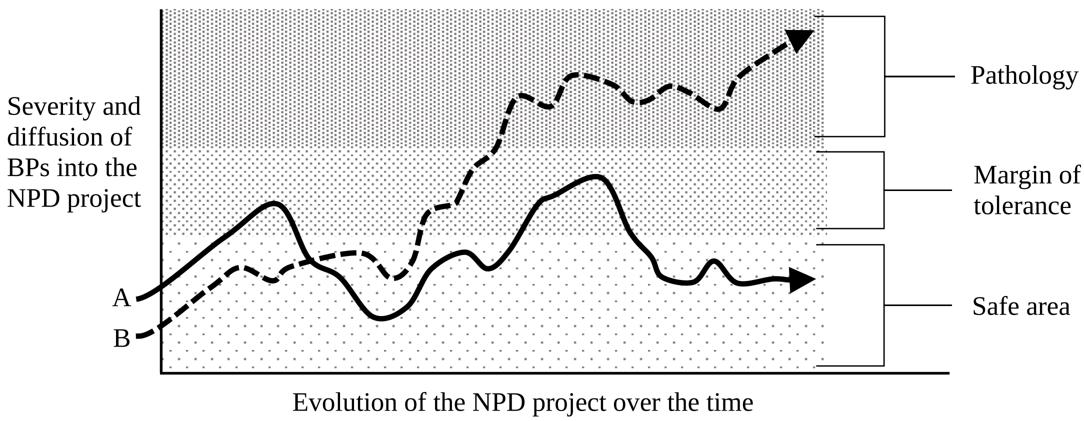


Figure 3 – BPs and Products' features