ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conferences IDETC 2017
August 6-9, 2017, Cleveland, Ohio, USA

DETC2017-68290

EXPLORING ARCHITECTURE SELECTION AND SYSTEM EVOLVABILITY

Samantha White
Graduate Research Assistant
North Carolina State University
Raleigh, NC, USA
sbwhite2@ncsu.edu

Dr. Scott Ferguson[1]
Associate Professor
North Carolina State University
Raleigh, NC, US
scott_ferguson@ncsu.edu

ABSTRACT
For engineered systems, one of the first decisions a designer must make is the architecture that will solve the established high level function. In most cases, this can be accomplished in a multitude of ways, with each original architecture having strengths and weaknesses. This paper explores how the architecture choice for a system impacts the ability to evolve and meet future needs. The lessons learned from this paper are extracted from a case study where three systems that perform the same task via different architectural solutions are considered. These systems are them compared to understand how well they adhere to, or violate, the Empirically-Derived Principles for Designing Products with Flexibility for Future Evolution introduced by Tilstra et al.

1 INTRODUCTION
Every problem can present multiple solutions. In the same sense, a function can be realized in multiple solution forms [1,2]. These larger architecture decisions form the framework for the rest of the engineered system. For example, a car and a horse-driven buggy accomplish the same task of transporting a chassis, but the supporting components have vastly different demands and properties. Requirements in these systems often change during the design process, but they can also change after the system has been put into the field [3]. While the system architecture can be changed during the design process, it is unlikely to change after the fact due to a high cost associated with doing so. Knowing that a system needs to change to meet future needs, is one of these base architectures, or core solution, more apt to evolve?

In *Design for X*, where *X* is *Evolvable Systems*, Tilstra et al. developed a set of empirically-derived principles for designing products with flexibility for future evolutions [4]. These principles, also known as *Guidelines for Product Flexibility*, provide insights through modularity, parts reduction, spatial, interface decoupling, and adjustability approaches to guide designers, but do not offer any specifics when choosing between multiple architectural options. This paper investigates the relationship between these principles and the selection of a system architecture by conducting a case study that compares a set of systems that have similar functionality but different architectures. These systems are analyzed to determine how closely they adhere to the guidelines and are then compared. Their ability to achieve future functionality based on the guidelines are discussed to determine if a primary architecture is inherently better for evolvability. This research combines previous research in both Architecture Option Theory [5,6] and the guidelines for product evolvability [4].

2 BACKGROUND
A system that evolves can add increased value to stakeholders by accomplishing new engineering criteria, operating in a new environment, and extending service life. Major concepts often associated with system evolution are modularity [7–11], architecture, and excess.

Baldwin, et al describes a modularized product as being divided according to a formal architecture or plan "to make complexity manageable, enable parallel work, and accommodate future uncertainty" [12]. Systems can be classified by their levels of modularity and granularity [13] or quantified by the modularization function [14] . Modularity also plays a role in *Design for the Life Cycle* [15], which can be considered an evolution if the service life is extended beyond initial requirements. While the appropriate division of modules is essential to evolvability, the interface complexity between modules also plays a key role in future evolutions [16].

Ulrich describes product architecture as "the scheme by which the function of a product is allocated to physical components" and identifies a linkage from product architecture to product change [17]. Siddiqi and de Weck later addressed a system architecture's role in reconfigurable systems through an analysis of 33 systems [18]. These systems were categorized by

[1]Corresponding Author

their modularity, and thusly, their reconfigurability. Architecture option theory was introduced as a basis for design for adaptability [19] by Engel and Browning [5]. This approach aimed to maximize system life time value by extending the life of systems via adaptability. This method was based in economic principles, financial options theory and transaction cost theory, to select the architecture for optimal lifetime value. In 2015, this concept was refined and tested in six industrial case studies to demonstrate a reduction in system's lifetime cost, reduction in systems' upgrade cycle-time, and an increase in systems' lifespan [6]. In 2016, these concepts were further refined to incorporate a measure of a products "*architecture adaptability value* (AAV)" from four industrial case studies [20]. These architectural models involved regrouping modules and altering components with great success. However, in all the systems, the broad, architectural solution (such as car vs. horse-drawn buggy) had already been chosen. These works guide designers in an arrangement of a final, more adaptable architecture with the selected core concept.

Excess is defined as "the quantity of surplus in a system once the necessities of the system are met." Prior research by Tackett et al. analyzed 210 engineered systems and concluded that system excess and modularity is critical to evolve an in-service complex engineered system [21]. Strategically placed excess can allow a system to evolve while only needing to change a few components or modules. Evolvability can be quantified based on the excess capacity of a system [22]. To visualize where a system might have excess that can be useful in future evolutions, excess in a system can be mapped. The method presented by Cansler et al. is based on a functional decomposition approach to provide designers with a visual tool to locate potential bottlenecks (areas that prevent future evolutions) in systems [23]. The result is a component map that shows and quantifies the actual and maximum possible flows between components. The relationship between excess and system evolution was shown via a stress-testing approach by Cansler et al., that used an excess map of a nerf gun to test the system against possible future evolutions [24]. While the mapping system shows designers where excess exists for future evolutions, certain architectural decisions must have already been made to formulate one. This is most useful when you have a product that needs evolving, but does not provide the same value when deciding between architectural concepts for an initial system.

The *Guidelines for Product Flexibility for Future Evolution* were originally introduced in 2007 ASME by Keese et al [25], and later expounded on in 2013 by Tilstra et al [4]. The principles were the result of two merged studies: a patent study of flexible products and a study of consumer products analyzed with a product flexibility matrix. These empirically-derived principles were created to guide the design process towards a more flexible, and thus evolvable, system. These 24

Table 1. Guidelines for Product Flexibility [4]

Guidelines for Product Flexibility for Future Evolution		
Modularity Approach	1	Using separate modules to carry out functions that are not closely related
	2	Confining functions to single modules
	3	Confining functions to as few unique components as possible
	4	Dividing modules into multiple smaller, identical modules
	5	Collecting parts which are not anticipated to change in time into separate modules
	6	Collecting parts which perform functions associated with the same energy domain into separate modules
Parts Reduction Approach	7	Sharing functions in a module or part if the functions are closely related
	8	Using duplicate parts as much as possible without raising part count
Spatial Approach	9	Creating room on the exterior surfaces of the device, around interior modules, and around those parts which are designed to interface with humans
	10	Providing free interfaces and expansive, unobstructed surfaces for new interfaces
	11	Extending the available area on the transmission components of the device
	12	Locating those parts which are anticipated to change near the exterior of the device
	13	Reducing nesting of parts and modules
Interface Decoupling Approach	14	Standardizing or reducing the number of different connectors used between modules
	15	Reducing the number of fasteners used, or eliminating them entirely
	16	Reducing the number of contact points between modules
	17	Simplifying the geometry of modular interfaces
	18	Routing flows of energy, information and materials so that they are able to bypass each module at need
	19	Creating detachable modules
	20	Using a framework for mounting multiple modules
	21	Using compliant materials
	22	Simplifying the geometry of each component
Adjustability Approach	23	Controlling the tuning of design parameters
	24	Providing the capability for excess energy storage or importation

guidelines are organized into five approach categories: modularity approach, parts reduction approach, spatial approach, interface decoupling approach, and adjustability approach. The full set of guidelines are shown in Table 1.

These guidelines were demonstrated with various products that clearly demonstrate each point, and then used to guide the design of a guitar string changer. However, the full list of guidelines was not used to compare multiple systems of varying architectures. The case study presented in this paper answers the following questions related to the guidelines.

- When faced with multiple architectural options, how do these guidelines aid selection?
- When do the guidelines ever contradict each other?
- And, what is the relationship between the number of guidelines followed and the relative evolvability of a product in comparison to a counterpart?

3 RESEARCH METHODOLOGY

To better understand and test the *Guidelines for Product Flexibility for Future Evolutions*, a system comparison was conducted. This comparison used systems that accomplish the same base functions with different architectures to compare how closely they follow the proposed guidelines. Because there is no way to definitively say if these products will evolve in the future, potential future evolution paths will be identified and historical data will be used to determine if these guidelines lead to a more evolvable base system architecture. The steps of this product comparison are outlined below.

Step 1. Select systems that accomplish the same base function with different methodologies.

The base function of the system is the two word phrase that defines what the system does. This could be something like “transport passengers”, “provide light”, or “cut grass”. The systems selected will be of similar scale and price range, such as a lamp and oil lantern as opposed to a match and stadium lighting. While a match and stadium lighting both perform the same function of “provide light”, they do not make for an adequate comparison, as both would not be in discussion for the same design task. The differences in the systems must extend beyond brand and minor component design to the way it accomplishes the task. The greater system differences allows a greater breath of understanding of how these principles would apply to greatly varying systems.

Step 2. Acquire the systems.

Product acquisition allows the opportunity to fully understanding the systems. The systems will be dissected n following steps for an even greater understanding [2,26].

Step 3. Generate a list of customer needs from current operating capabilities of the system.

The customer needs can be realized from the manufacturers’ specifications of the product. These include system performance, additional functionalities, and operating conditions.

Step 4. Compare the list of needs to identify possible future evolutionary paths.

These varying needs can provide insight to what a future evolution of these systems might be. Currently, a consumer must make tradeoffs between needs when purchasing a product. A company may decide to encompass some of the additional needs to make their system more competitive in the marketplace. Possible future evolution paths can be identified as bringing all systems’ capabilities and functions up to the current best of the collection of customer needs. By identifying possible future evolution paths, the systems can be judged on their potential to meet them.

Step 5. Decompose the systems into components while recording their configuration and spacing.

Decomposing the systems reveals interfaces between components as well as internal packaging and configurations. These properties, as well as the functionality of modules, is critical in determining if a system does or does not follow a guideline. Components will be tabulated, and internal configurations will be photographed.

Step 6. Create a DSM and function flow map for each system.

These product decomposition tools show which components affect each other and how flows are passed within the system [27,28]. While these are not directly needed for any specific guideline, they are useful in understanding component relationships that will aid in determining if a guideline is or is not followed.

Step 7. Characterize how each of these systems relates to the guidelines.

For each system, the full list of guidelines will be compared to its current configuration. If a guideline is definitively followed or defied, it will be noted. For guidelines that could be argued either way or that are completely non-applicable, it will be passed over. Since this is a set of guiding principles and not rules, it is assumed that not every guideline will be applicable, useful, or definable for every system.

Step 8. Analyze the defied guidelines. Determine possible system improvements.

Each violated guideline for a system will be analyzed. Determine what, if any, improvements could be made to the system to follow the guideline, or if the guideline would not aid the system’s evolvability in that case and why.

Step 9. Rank systems based on number of guidelines followed or not followed.

If a clear ranking appears, rank the systems. A clear ranking would be the systems with the most number of guidelines followed also break the least number of guidelines, and this pattern continued with the second and third most followed/least violated.

Step 10. Assess whether these guidelines make a more evolvable system using possible future evolutions and historical data if appropriate.

Ease of reaching a possible future evolution can be determined by excess embedded in the system, in either component capabilities or space to add additional modules. Historical data can come in the form of previous generations of a product or an extended product line based off the original architecture.

4 CASE STUDY

This section follows the established steps 1-10 to answer the questions posed at the end of section 2.

Steps 1-2. Select systems that accomplish the same base function with different methodologies.

For the case study, three automatic ball launchers were selected. All three systems are designed for dogs to play fetch without human intervention. The three systems analyzed are the iFetch Too (Fig. 1A) [29], the PetSafe Ball Launcher (Fig. 1B) [30], and the GoDogGo G4 Launcher (Fig. 1C) [31]. These systems are similar in price, $199.99, $149.99, and $199.99 respectively, and perform the same top level function: "launch ball". They are also similar in scale, as they launch regulation sized tennis balls and are made to be relatively portable. However, they use different internal architectures to accomplish this task which makes them ideal for a case study. Two use potential spring energy to drive a kicker that hits the ball: one vertically mounted with 360° of movement, and one horizontally mounted without full circular movement. The third uses oppositely spinning wheels that the ball is squeezed through to gain momentum. These systems were broken down to further study and understand their internal workings.

Fig. 1A- iFetch Too Ball Launcher

Fig. 1B- PetSafe Ball Launcher

Fig. 1C- GoDogGo G4 Ball Launcher

Step 3. Generate a list of customer needs from current operating capabilities of the system.

Based on manufacturers' specifications for the systems, a list of customer needs were created and are shown in Table 2. These needs encompass the specifications surrounding the function "launch ball" as well as additional features of the systems like adjustable launch angles and distances.

Step 4. Compare the list of needs to identify possible future evolutionary paths.

While these three systems perform the same function, they boast a few different features. Based on the created list of customer needs, the features that the systems did not share were extrapolated and tabulated. These dissimilar system traits are shown in Figure 2. Each feature that the system does not currently have, represents a possible future evolution path because it was deemed valuable enough to customers for another company to include in their systems.

Table 2. Customer Needs for Three Launch Systems

iFetch	PetSafe	GoDogGo
• Must be able to operate without a cord (rechargeable battery) • Must launch ball at 10, 25, and 40 feet on set settings with the option for a variable, random distance • Must launch standard size tennis balls • Must be small enough to be portable • Must be reliable • Must have all moving mechanisms internal to the machine as to not injure a dog • Must have large catch basin that is low enough for most dog breeds to reach	• Must be able to operate with or without a cord (plug in option or 6 'D' cell batteries) • Must launch ball between 8-30 feet • Must have 9 different distance settings • Must have 6 angle settings • Must launch different ball sizes and textures • Mandatory 15 minute rest breaks • Must have safety motion sensor in front to insure no damage to dogs/people • Must be small enough to be portable • Must be reliable • Must have all moving mechanisms internal to the machine as to not injure a dog • Must have large catch basin that is low enough for most dog breeds to reach	• Must operate with or without a cord (6 'D' cell batteries) • Must be able to launch between 9' and 45' based on machine setting (3 settings) and ball type • Must launch different ball sizes and textures • Must launch automatically on time intervals or by remote control • Must not activate kicker/launcher mechanism without ball in chute • Must be small enough to be portable • Must be reliable • Must have all moving mechanisms internal to the machine as to not injure a dog • Must have large catch basin that is low enough for most dog breeds to reach

	iFetch	PetSafe	GoDogGo
Rechargeable batteries	x		
Safety sensor in front of the machine		x	
Mandatory rest periods		x	
Can launch different size balls		x	x
Random distance mode	x		
Holds multiple balls			x
Can change time delay on launch			x
Additional remote control operation			x
Carrying handle	x	x	
Ability to change launch angle		x	

Fig 2. Varying Features Among Three Ball Launch Systems

It is important to note here that a system that can already reach an operating criterion cannot be considered evolvable by meeting this same criterion at a future state. For example, the iFetch system operates by a rechargeable battery, while the PetSafe and GoDogGo must be tethered to the wall or have 6 D cell batteries added to operate. The PetSafe and GoDogGo could evolve to run off rechargeable batteries and be considered evolvable in this sense, but the iFetch would not because it already started in this more desirable operating state. For this reason, *Design for Evolvability* cannot be a sole objective when designing a system. Other design objectives, such as *Design for Manufacturability* or *Design for Sustainability,* are not considered in this case study. The most evolvable systems would have limited features with the ability to achieve them later and not what a consumer would want. Therefore, evolvability must be a compromise with other design objectives that bring value to a system.

Step 5. Decompose the systems into components while recording their configuration and spacing.

The systems were dissected into their components. Because these are relatively small systems, most components could be listed individually. However, a module that would be replaced as a whole on future evolutions is considered a single component. An example of this is the motors that drive the systems. The motor has additional, individual components that comprises it, but if higher RPM or torque is required, a user or manufacturer would likely use a larger stock motor than altering the existing one. The created component lists for the systems are shown in table 3.

Table 3. Ball Launcher Components

iFetch	PetSafe	GoDogGo
• Case • Ball Channel • Wires • 2 Motors • 2 Wheels • Battery • Circuit Board • Power/Distance Switch • Safety Limiter Switch • Ball Pressure Sensor • Ball Release Lever	• Case • Ball Channel • Circuit Board • Wires • Kicker • Spring • Motor • Angle Board • Distance Knob • Angle Knob • Angle Rack • Distance Rack and Release Column • Release Lever • Locking Gear • Power Source (Batteries/Wall Plug) • Power Switch • Safety Sensor • Speaker • Ball Sensor	• Bucket • Case • Ball Pressure Sensor • Wires • Power Switch • Power Source (Batteries/Wall Plug) • Circuit Board • Motor • Spring • Kicker

The iFetch propels the ball by two oppositely spinning wheels driven by two motors. The human operator controls the system from a single button and slider in the back. The button is compressed once to turn the system on its low setting, a second time for the mid-range, a third time for the high setting, a fourth time for a different random launch distance with each ball, and a fifth time to turn off. The slider can put the system in a safer "indoor" mode that prevents the system from going above its low setting. The canine operator drops the ball in the opening in the top of the system. A pressure switch activates if the system is on, accelerates the wheels, and drops the ball through the chute to fly out the front opening. The system is powered by a rechargeable battery or wall plug. All control systems are purely electrical, and the internal operating components of this system are contained in close-fitting plastic. The internal layout of the system can be seen in Fig 3.

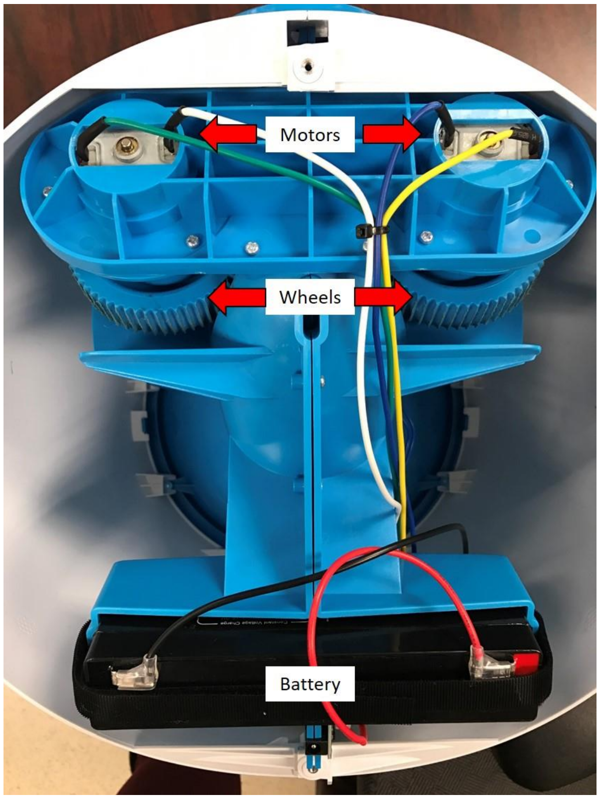

Fig. 3 Internal Components of iFetch Too

The PetSafe ball launcher uses a motor to pull a spring and kicker backwards on a horizontal tilted plate. The spring will go back until the release lever is reached, and the kicker will fly forward from the force in the spring, hitting the ball, and propelling it out of the front. The human operator can set the machine at nine distant settings and six angle settings by turning two knobs on the exterior of the system. The knobs are connected to gears that drive internal racks. The distance setting changes the location of the release lever, consequently making the spring stretch different lengths. The angle setting changes the angle of the plate that the ball and components sit on. The system is turned on and off by a power switch. The canine operator drops the ball in the top opening of the machine. Once the ball is sitting in the firing chamber, it activates the beam break sensor telling the motor to start pulling the spring. However, the PetSafe launcher has a built in safety feature that will not launch the ball if it senses motion up to 7 feet in front of the system to prevent injuries to humans and dogs. It also has a programmed in 15-minute interval rest feature to keep dogs from overexertion. The system is powered by either 6 D cell batteries or a wall plug. Control systems are a mix of mechanical and electrical, and the internal components are mounted either openly to the angle board or to the case itself without any tight fitting coverings. The internal layout of the system can be seen in Fig 4.

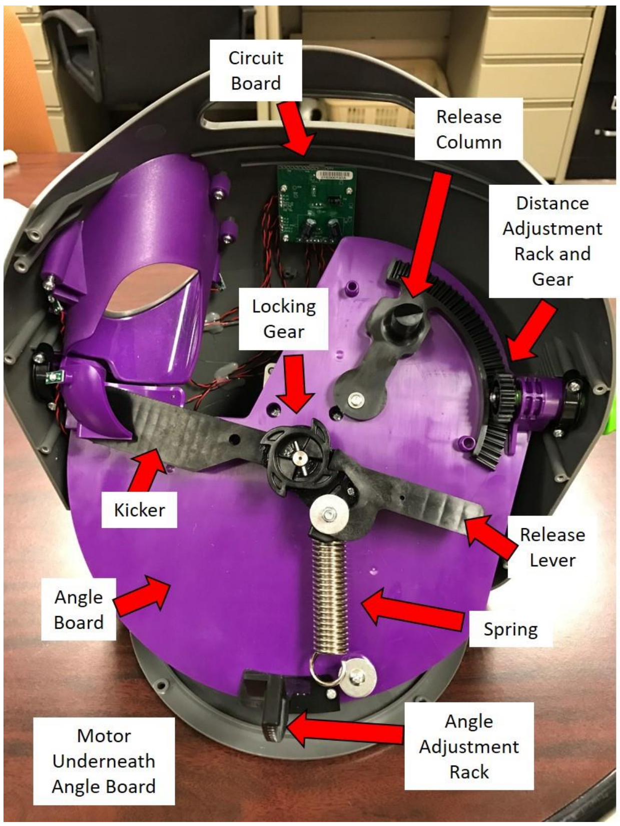

Fig. 4 Internal Components of PetSafe Launcher

The GoDogGo G4 also uses a motor to pull a kicker straight up against a spring, and the kicker releases when it passes vertical and is pulled down by the spring. The kicker follows a full circular path and hits the ball out of the front holding ramp. The human operator changes launch distance by manually stretching the spring into different holding slots at the bottom of the system. The system can hold multiple balls at once, and the human user can change time between launches by either a button on the rear of the machine or a handheld remote. This button and remote can also put the system in auto-launch mode. The canine operator would put the ball in the top bucket, and the ball would follow the ramp down into the firing chamber. The ball depresses a pressure switch telling the motor to begin pulling the spring and fires the ball. The system is powered by either 6 D cell batteries or a wall plug. Control systems are a mix of mechanical and electrical, and the internal components are mounted to the base without and coverings. The internal layout of the system can be seen in Fig 5.

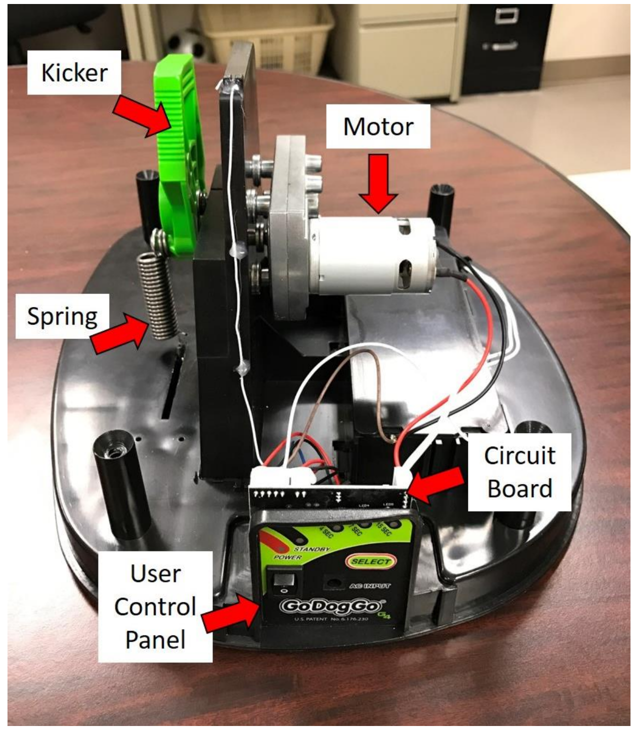

Fig. 5 Internal Components of GoDogGo G4

Step 6. Create a DSM and function flow map for each system.

Since many of the guidelines are based on system modules, part numbers, and interfaces, DSMs were created for each system to show part interactions. Additionally, functional diagrams were created of these systems to show how energy, materials, and signals were passed through components. In future evolutions, a single component may need to be changed to meet a new need. These two diagrams for each system can show how this will affect other components in the system, and how that might prevent a system from easily evolving.

In all three systems, the circuit board acts as the central hub of functionalities, sending signals to different components during operation. The two spring driven systems, the PetSafe and GoDogGo, also share similarities in the motor to spring and kicker portion of the functional diagram because these diagrams do not model component orientation. These diagrams helped determine feasibility of future evolutions in step 8 by showing what flows must interface with each component. If changing a component prevents it from accepting or passing a vital flow, additional components must be added or changed to accommodate it. These figures are shown in the Appendix.

Step 7. Characterize how each system relates to the guidelines.

Using the component lists, component layouts, DSMs, and functional models, each launcher system was examined against the 24 empirically derived principles for product flexibility. The guidelines that the systems followed are shown in Table 4 and the guidelines that the systems did not follow are shown in Table 5.

Table 4. Guidelines Followed in Ball Launch Systems

System	Guideline
All Systems	7- Sharing functions in a module or part if the functions are closely related 10- Providing free interfaces and expansive, unobstructed surfaces for new interfaces 20- Using a framework for mounting multiple modules 24- Providing the capability for excess energy storage or importation
iFetch	3- Confining functions to as few unique components as possible 8- Using duplicate parts as much as possible without raising part count 16- Reducing the number of contact points between modules 17- Simplifying the geometry of modular interfaces
PetSafe	8- Using duplicate parts as much as possible without raising part count
GoDogGo	3- Confining functions to as few unique components as possible 9- Creating room on the exterior surfaces of the device, around interior modules, and around those parts which are designed to interface with humans 13- Reducing nesting of parts and modules 16- Reducing the number of contact points between modules 17- Simplifying the geometry of modular interfaces 19- Creating detachable modules

Guidelines Followed by All Systems

The three systems share some similarities in following guidelines that would aid them in future evolutions. First, all three have a circuit board that acts as the central hub for all control processes. This single module controls multiple, closely related functions (guideline 7). They also all have additional unobstructed spaces to potentially add components later (guideline 10). This includes the entire outer surface of the systems' cases and empty space within. A designer could add modules for additional functionality with minimal concern about space availability. The hollow outer cases create a framework to mount all modules (guideline 20). This reduces the need to create packaging for any current or future components, as well as makes these components easier to access once inside the framework. Finally, all systems contain some form of excess energy storage or importation (guideline 24). Research by Tackett previously identified excess as a key to evolvability of engineered systems [21]. In the PetSafe and GoDogGo systems, this excess energy is contained in the spring that drives the kicker. The springs can be stretched further, giving the system a greater output. In the same manner, the iFetch motor is not operating at maximum potential. By increasing the operating RPM of the motor, this system would also deliver a greater output of speed and distance a tennis ball is launched.

Guidelines Followed by iFetch

The components of the iFetch all accomplish a single task, and all functions of the system are generally constrained to one or very few modules (guideline 3). The ball is held in a single chute, the battery supplies power, the motor drives the wheels, and the wheels launch the ball. No function is distributed among multiple superfluous parts. The system also uses twin wheels and motors to launch the ball. The duplicate parts can be manufactured on the same line, and the parts can be redesigned together if they need to be upgraded (guideline 8). The modules in the system are only connected at single vital points to transfer energies/flows (guideline 16). These interfaces are simple electrical (soldering) or mechanical connections (guideline 17). If a component needs to be upgraded, it will then be more likely that the new piece can interface with the system and help prevent cascading changes being needed.

Guidelines Followed by PetSafe

The PetSafe launcher has two modes of launch adjustment: angle and distance. These manual control modules are made of the same gears, internal rods, and springs. These duplicate parts offer the same benefits as described for the iFetch (guideline 8).

Guidelines Followed by GoDogGo

The GoDogGo launcher boasts the least number of components, and functions are accomplished by as few components as possible (guideline 3). All internal components are mounted to the inside base without any additional casing. This free volume of space around each part allows changes to be made with less concern about spatial constraints (guideline 9). The pieces are also not nested in each other (guideline 13), with as few contact points as necessary to operate (guideline 16). The interactions between parts are also simple mechanical or electrical connections that could work with upgraded versions of those components (guideline 17). Most components are also made to be replaced if they wear out. GoDogGo Inc. sells the spring, kicker, and ball sensor separately for system repairs. These parts can all be detached from the system, as well as the bucket on top to accommodate shorter dogs (guideline 19). The ease of access to these pieces, as well as their ability to be removed and replaced, allows for simple system evolution by changing these components.

Table 5. Guidelines Not Followed in Ball Launch Systems

System	Guideline
All Systems	21- Using compliant materials
iFetch	9- Creating room on the exterior surfaces of the device, around interior modules, and around those parts which are designed to interface with humans 13- Reducing nesting of parts and modules 19- Creating detachable modules
PetSafe	3- Confining functions to as few unique components as possible 9- Creating room on the exterior surfaces of the device, around interior modules, and around those parts which are designed to interface with humans 13- Reducing nesting of parts and modules 16- Reducing the number of contact points between modules 17- Simplifying the geometry of modular interfaces 19- Creating detachable modules
GoDogGo	8- Using duplicate parts as much as possible without raising part count

Guidelines Not Followed by All Systems

The three systems use rigid components and packaging (guideline 21). These materials cannot adapt the way compliant materials do in future evolutions. However, all systems also use a framework to mount the modules (guideline 20) that makes the need for compliant materials less necessary, as the parts are not being fit into tight spaces, but mechanically mounted to the case.

Guidelines Not Followed by iFetch

The motors and wheels that launch the ball in the iFetch are incased in tight fitting plastic housing. This configuration makes it more difficult to change those components in the future, as a new design might not fit in the available space (guidelines 9 and 13). Additionally, these components are not easily detachable from the system (guideline 19). The battery is the only component that could be easily taken out, as it is secured in its slot with Velcro. However, the battery also sits in a form fitting plastic packaging that will prevent any battery evolution if a new battery would require a different footprint.

Guidelines Not Followed by PetSafe

The PetSafe launcher fails many of the guidelines based on its many components and their close proximity. Functions that require a single part in the other two systems require multiple parts in the PetSafe (guideline 3). Due to the close fittings of the gears and other moving mechanical components, there is not a lot of spare space surrounding these components for future evolutions (guideline 9). Some parts, like the knob and gear mechanisms and locking gear and kicker are nested together (guideline 13). The configuration of all the mechanical components on top of the angle board requires many contact points among components (guideline 16) and more complex geometric interfaces (guideline 17). The system is also not meant to be taken apart, and the modules are not easily detachable from each other or the case (guideline 19).

Guidelines Not Followed by GoDogGo

The GoDogGo launcher does not use any duplicate parts (guideline 8).

Step 8-9. Analyze the defied guidelines. Determine possible system improvements. Rank systems based on number of guidelines followed or not followed.

The three ball launch systems can be ranked based on how many guidelines they follow or defy.

#1 – GoDogGo

The GoDogGo follows the most guidelines (10) and defies the least (2). This is attributed largely to the fact that the system uses as few components as necessary, packages them with little interference to each other, and connects components when necessary with simple interfaces. The two guidelines it does not follow is using compliant materials and using duplicate parts. However, these guidelines are not entirely applicable to the system. The compliant materials aid future evolution by molding to the available space. The large, empty framework of the case means components do not have to fit snugly within it, but simply be mounted to the wall. The system also uses so few parts that duplicate parts would be redundant and unnecessary.

#2- iFetch

The iFetch follows the second most guidelines (8) and defies the second least (4). The system has a low part count and simple interfaces between components. The iFetch is ranked second largely due to the motor casings. These close fitting plastic pieces will prevent future evolutions of the motor to fit in those spaces. The plastic casings could be made larger or even changed to a design that does not fully encapsulate them. Additionally, the pieces of the system are not easily detachable. By changing some of the mounting components to easier to access screws, Velcro, or other removable attachments, the system would be considered more evolvable.

#3- PetSafe

The PetSafe follows the least guidelines (5) and defies the most (7). This can be mainly attributed to the relatively long parts list and the close fitting configuration of internal components. The system could eliminate some of the parts and simplify the geometric connections by changing the angle and distance control mechanisms. Currently the angle board mounts all the launch components and requires multiple components to change angles. This could be altered by simply changing the exit ramp angle instead of the internal components. The geometry would be simplified, and many components could be eliminated. Additionally, the gear and rack system could be changed to a simpler notched configuration or an electric control system, decreasing part count and embedding additional excess. The gear and rack system could also be changed in the distance settings to a manual slider of the release point or an electric control system. This would also decrease part count, simplify interfaces, and potentially be easier for a consumer to use. By changing the two control settings, there will be room internally because the angle board will no longer be necessary. Then, there will be excess room around parts, an improved framework for mounting components, and fewer contact points between modules.

Step 10. Determine if these guidelines make a more evolvable system using possible future evolutions and historical data if appropriate.

By comparing the features of the current systems from Fig. 2, possible future evolutionary paths can be identified for the launchers. The feasibility of these evolutions can be predicted based on the related guidelines. The features considered as future evolutions for each system are shown in Table 6.

Table 6. Possible Future Evolutionary Paths for Launch Systems

GoDogGo	iFetch	PetSafe
1. Change power system to rechargeable battery 2. Addition of safety sensor 3. Mandatory rest periods 4. Random distance mode 5. Carrying handle 6. Variable launch angle	1. Addition of safety sensor 2. Mandatory rest periods 3. Ability to launch different sized balls 4. Ability to hold multiple balls 5. Time delay launch option 6. Remote control operation 7. Variable launch angle	1. Change power system to rechargeable battery 2. Random distance mode 3. Ability to hold multiple balls 4. Time delay launch option 5. Remote Control Operation

Based on the functionalities that are missing from the current GoDogGo system, possible evolutions include a rechargeable battery, a safety sensor, mandatory rest periods, a random distant mode, a carrying handle, and the ability to change launch angle. The GoDogGo follows guideline 7, sharing functions in a module or part if the functions are closely related, with its control system in the circuit board. Because of this, new control modes can be easily realized, such as mandatory rest periods. The unimpeded space on the exterior of the system and around components, as specified by guidelines 9 and 10, allow for additional modules to be attached or select parts to be upgraded. This would allow the system to incorporate a rechargeable battery, a carrying handle, a safety sensor, and a variable launch angle. The launch distance of the system is set by manually stretching the spring into a notch. While the spatial guidelines may allow this spring to stretch farther to achieve a greater launch distance, there is not an easy path to achieve a random launch angle. Of the GoDogGo's 6 proposed evolutions, the guidelines lend the system to achieve 5 of them.

Proposed future evolutions for the iFetch include a safety sensor, mandatory rest periods, the ability to launch different sized balls, the ability to hold multiple balls, the option to time delay ball launch, remote control operation, and the ability to change launch angle. The iFetch also follows guideline 7, sharing functions in a module or part if the functions are closely related, with its control system in the circuit board. Because of this, the iFetch can be programmed for mandatory rest periods, the option to time delay ball launch, and accept remote control operation. The external free space built into the system from guideline 10 allows additional modules to be mounted such as a safety sensor, a ball holding chamber, and an external prop to change system angle. However, the iFetch does not follow guidelines 9, creating room around interior modules, and 13, reducing nesting of parts and modules, with its dual motors and wheels encapsulated in tight plastic fittings. Because of this, the wheels are not easily moved to launch different sized balls. Due to the architecture, smaller balls slide through the wheels and do not make contact with them to gain momentum to propel them forward. Larger balls get stuck in the ball chute because they cannot fit through the wheels. Of the iFetch's 7 proposed evolutions, the guidelines lend the system to achieve 6 of them.

Future evolutions of the PetSafe include a rechargeable battery, a random distance mode, the ability to hold multiple balls, the option to time delay ball launch, and remote control operation. The PetSafe launcher also follows guideline 7, sharing functions in a module or part if the functions are closely related, with its control system in the circuit board. Because of this, the option to time delay ball launch and remote control compatibility can be added. The free interfaces and expansive, unobstructed surfaces for new interfaces as provided by guideline 10 allows the ability to add a rechargeable battery. While this might also allow the addition of a ball holding chamber on top to hold multiple balls, the system currently does not have a component in place to hold the ball before launch. The launch is controlled by a beam brake sensor in a single ball sized plastic cradle. Because there is not internal space around these modules, as the system broke guideline 9, the functionality must be added externally. This would include a chamber, a ball releasing latch, and a second ball sensor. This evolution would duplicate many components, and make the system larger and more expensive than necessary. Additionally, since the launch distance is controlled by a manual knob moving the release lever, the random distance setting cannot be easily realized. Of the PetSafe's 5 proposed evolutions, the guidelines lend the system to achieve 3 of them.

In addition to the future proposed guidelines, historical data can be used to analyze the evolutionary capability of the system. The GoDogGo has both a scaled down version of its system in its current product line and multiple previous generations of its system. The product line includes a similar style machine that is intended to launch miniature tennis balls for smaller dogs and uses some of the same components of the GoDogGo G4 launcher, such as the control system. Currently, the iFetch systems have extended the product line to create three different launch systems. One purely gravity driven system that does not share any components with the iFetch Too, and the original iFetch that is the same mechanical workings of the iFetch Too, only scaled down to launch miniature sized tennis balls. The iFetch and iFetch Too share multiple components, like the control system. The PetSafe launcher is the first generation ball launcher by PetSafe and does not have any extensions to the product line.

The three systems all have the same anticipated future needs: a scaled down version for smaller dogs, a version that increases the base functionality of the original system (launching balls farther, launching different types and sizes of balls, etc.), adding additional functionality to the system beyond launch ball (addition of dog treat dispenser, addition of voice recording and speakers, etc), and extending product life. Based on the guidelines, the rankings of the three systems support both potential future evolutions and historical data. The GoDogGo launcher is the system that would likely be most successful in evolution based on the guidelines. It is also the system that has undergone the most number of generational improvements. The iFetch began as a launcher for small scale tennis balls, and has evolved to the iFetch Too that was used in this study to support play for full size tennis balls and larger dogs. Finally, the PetSafe followed the fewest guidelines, and it has not had any evolutions to date.

5 CONCLUSIONS AND FUTURE WORK

The case study analyzed three separate architectures that were driven from one of two energy domains: the potential energy from a stretched spring (two systems) or the mechanical energy powering oppositely spinning wheels (one system). Prior to component arrangement, one of these two energy domains had to be selected in the initial design process. The question to be asked in *Design for Flexibility* is, "is one of these architectures inherently more evolvable than the others?" As Ulrich described architecture as "the scheme by which the function of a product is allocated to physical components" [17], we can describe the scheme in a broad sense as the driving component of the system, that is, spring or motor driven wheels. As seen from the case study, two different spring driven systems were ranked as both the most evolvable and least evolvable. Therefore, neither architecture - spring or motor driven wheels - is inherently more evolvable than the other. The results of the case study were that the initial architecture did not influence evolvability as much as how that architecture is implemented to incorporate excess in key, nested components and embody the principles defined by Tilstra et al. Architectures should be selected that can be constructed in branched function diagrams, so modules and components can be replaced with minimal negative interactions. Components that emerge as central hubs for flows should be designed with excess incorporated in them to prevent these components from bottlenecking future evolutions.

If the primary decision of "what method do we use to serve a function" is not the primary driver of evolvability, the decisions of how to allocate those functions to parts and compose them in a system are. Architectures that follow the guidelines are more appropriate for future evolutions and are also likely a result from past evolutions. These principles are intended to function as high level guidelines as oppose to strict laws to enforce. It is unlikely that every guideline will be applicable to a system.

The methodology presented provided a way to compare the potential evolvability of systems with varying architectural layouts. The rankings produced from the procedure largely supported the guidelines based on future evolutionary paths and historical data. However, it also revealed situations where following guidelines might not necessarily be constructive to the evolvability of a system or is in direct competition with another guideline. While the methodology was a useful tool in demonstrating evolvability, it did not factor in customer satisfaction with a product or system value. Because design for evolvability is only a small subsector of the goals of design, these criteria could be considered in future iterations of the procedures to show how they are also, if at all, affected by the guidelines. Additional future work could extend this case study to more similar functioning systems with different architecture to fully encompass the list of guidelines, as some of the guidelines on the list were not appropriate for the systems studied.

ACKNOWLEDGMENTS

The authors gratefully acknowledge support from the National Science Foundation through NSF Grant No. CMMI-1301247 and the National Science Foundation Graduate Research Fellowship. Any opinions, findings, and conclusions presented in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] Weber, R. G., and Condoor, S. S., 1998, "Conceptual design using a synergistically compatible morphological matrix," Front. Educ. Conf. 1998. FIE'98. 28th Annu., **1**, pp. 171–176.

[2] Simpson, T. W., and Thevenot, H. J., 2005, "Using Product Dissection to Integrate Product Family Design Research Into the Classroom and Improve Students' Understanding of Platform Commonality," Vol. 5a 17th Int. Conf. Des. Theory Methodol., (814), pp. 375–382.

[3] Shankar, P., Morkos, B., and Summers, J. D., 2012, "Reasons for change propagation: A case study in an automotive OEM," Res. Eng. Des., **23**(4), pp. 291–303.

[4] Tilstra, A. H., Backlund, P. B., Seepersad, C. C. C., and Wood, K. L., 2015, "Principles for designing products with flexibility for future evolution," Int. J. Mass Cust., **5**(1), pp. 22–54.

[5] Engel, A., and Browning, T. R., 2008, "Designing systems for adaptability by means of architecture options," Syst. Eng., **11**(2), pp. 125–146.

[6] Engel, A., and Reich, Y., 2015, "Advancing architecture options theory: six industrial case studies," Syst. Eng., **18**(4), pp. 396–414.

[7] Jung, S., and Simpson, T. W., 2017, "New modularity indices for modularity assessment and clustering of product architecture," J. Eng. Des., **28**(1), pp. 1–22.
[8] Bonvoisin, J., Halstenberg, F., Buchert, T., and Stark, R., 2016, "A systematic literature review on modular product design," J. Eng. Des., **27**(7), pp. 488–514.
[9] ElMaraghy, H., and AlGeddawy, T., 2015, "A methodology for modular and changeable design architecture and application in automotive framing systems," J. Mech. Des. Trans. ASME, **137**(12).
[10] Otto, K., Hölttä-Otto, K., Simpson, T. W., Krause, D., Ripperda, S., and Moon, S. K., 2016, "Global Views on Modular Design Research: Linking Alternative Methods to Support Modular Product Family Concept Development," J. Mech. Des., **138**(7), p. 71101.
[11] Liu, C., Hildre, H. P. P., Zhang, H., and Rolvag, T., 2016, "Product architecture design of multi-modal products," Res. Eng. Des., **27**(4), pp. 331–346.
[12] Baldwin, C. Y., and Clark, K. B., 2004, "Modularity in the design of complex engineering systems," Complex Eng. Syst., pp. 175–205.
[13] Chiriac, N., Hölttä-Otto, K., Lysy, D., Suh, E., Hölttä-Otto, K., Lysy, D., and Suk Suh, E., 2011, "Level of Modularity and Different Levels of System Granularity," J. Mech. Des., p. 133: 101007.
[14] Mikkola, J. H., and Gassmann, O., 2003, "Managing Modularity of Product Architectures: Toward an Integrated Theory," IEEE Trans. Eng. Manag., **50**(2).
[15] Newcomb, P. J., Bras, B., and Rosen, D. W., 1998, "Implications of Modularity on Product Design for the Life Cycle," J. Mech. Des., **120**(3), p. 483.
[16] Beesemyer, J. C., Fulcoly, D. O., Ross, A. M., and Rhodes, D. H., 2011, "Developing Methods to Design for Evolvability: Research Approach and Preliminary Design Principles," 9th Conference on Systems Engineering Research, Los Angeles, CA, CA.
[17] Ulrich, K., 1995, "The role of product architecture in the manufacturing firm," Res. Policy, **24**, pp. 419–440.
[18] Siddiqi, A., and de Weck, O., 2008, "Modeling Methods and Conceptual Design Principles for Reconfigurable Systems," J. Mechnical Des., **130**, p. 101102–1 to 15.
[19] Kasarda, M. E., Terpenny, J. P., Inman, D., Precoda, K. R., Jelesko, J., Sahin, A., and Park, J., 2007, "Design for adaptability (DFAD)-a new concept for achieving sustainable design," Robot. Comput. Integr. Manuf., **23**(6), pp. 727–734.
[20] Engel, A., Browning, T. R., and Reich, Y., 2016, "Designing Products for Adaptability: Insights from Four Industrial Cases," Decis. Sci.
[21] Tackett, M., 2013, "A Mathematical Model for Quantifying System Evolvability using Excess and Modularity."
[22] Tackett, M. W. P., Mattson, C. a., and Ferguson, S. M., 2014, "A Model for Quantifying System Evolvability Based on Excess and Capacity," J. Mech. Des., **136**(5), p. 51002.
[23] Cansler, E. Z., White, S. B., Ferguson, S. M., and Mattson, C. A., 2016, "Excess Identification and Mapping in Engineered Systems," J. Mech. Des., **138**(8), p. 81103.
[24] Cansler, E. Z., Ferguson, S. M., and Mattson, C. A., 2015, "Exploring the Relationship Between Excess and System Evolutions Using a Stress-Test," Volume 7: 27th International Conference on Design Theory and Methodology, ASME, p. V007T06A041.
[25] Keese, D. A., Tilstra, A. H., Seepersad, C. C., and Wood, K. L., 2007, "Empirically-Derived Principles for Designing Products With Flexibility for Future Evolution," Volume 3: 19th International Conference on Design Theory and Methodology; 1st International Conference on Micro- and Nanosystems; and 9th International Conference on Advanced Vehicle Tire Technologies, Parts A and B, ASME, pp. 483–498.
[26] Moore-Russo, D., Grantham, K., Lewis, K., and BATEMAN, S. M. ., 2010, "Comparing Physical and Cyber-Enhanced Product Dissection: Analysis from Multiple Perspectives.," Int. J. Eng. Educ., **26**(6), pp. 1378–1390.
[27] Browning, T. R., Co, L., and Worth, F., 2001, "Applying the design structure matrix to system decomposition andintegration problems: a review and new directions," IEEE Trans. Eng. Manag., **48**(3), p. 48(3): 292-306.
[28] Tilstra, A. H., Seepersad, C. C., and Wood, K. L., 2012, "A high-definition design structure matrix (HDDSM) for the quantitative assessment of product architecture," 12, **23**(10–11), pp. 767–789.
[29] IFetch, "iFetch Too (Launches Standard Size Tennis Balls)," iFetch Shop.
[30] PetSafe, "Automatic Ball Launcher," PetSafe Store.
[31] GoDogGo, "GoDogGo Fetch Machine," GoDogGo Prod.

ANNEX A

LAUNCH SYSTEM DSMS

	Case	Ball Channel	Wires	Motors	Wheels	Battery	Circuit Board	Power/Distance Switch	Safety Limitor Switch	Pressure Sensor	Ball Release Lever
Case		1		1	1	1	1	1	1	1	1
Ball Channel										1	1
Wires				1		1	1	1	1		1
Motors			1		1						
Wheels											
Battery			1	1			1				
Circuit Board			1	1							
Power/Distance Switch			1				1				
Safety Limitor Switch			1								
Pressure Sensor			1								
Ball Release Lever											

Fig. 6 iFetch Too DSM

	Case	Ball Channel	Circuit Board	Wires	Kicker	Spring	Motor	Angle Board	Distance Knob	Angle Knob	Angle Rack	Distance Rack & Release Column	Release Lever	Locking Gear	Power Source	Power Switch	Safety Sensor	Speaker
Case		1	1	1				1	1	1	1					1	1	1
Ball Channel																		
Circuit Board				1			1											1
Wires			1				1											1
Kicker																		
Spring					1		1											
Motor					1	1												
Angle Board					1	1	1					1	1	1				
Distance Knob												1						
Angle Knob											1							
Angle Rack								1										
Distance Rack & Release Column						1							1					
Release Lever														1				
Locking Gear					1	1							1					
Power Source			1	1			1										1	1
Power Switch			1	1														
Safety Sensor			1	1														
Speaker																		

Fig. 7 PetSafe DSM

	Bucket	Case	Ball Sensor	Wires	Power Switch	Power Source	Circuit Board	Motor	Spring	Kicker
Bucket		1								
Case	1		1		1		1	1	1	1
Ball Sensor				1			1			
Wires							1	1		
Power Switch				1			1			
Power Source				1			1			
Circuit Board				1				1		
Motor									1	1
Spring								1		1
Kicker										

Fig. 8 GoDogGo DSM

ANNEX B

LAUNCH SYSTEM COMPONENT-FLOW DIAGRAMS

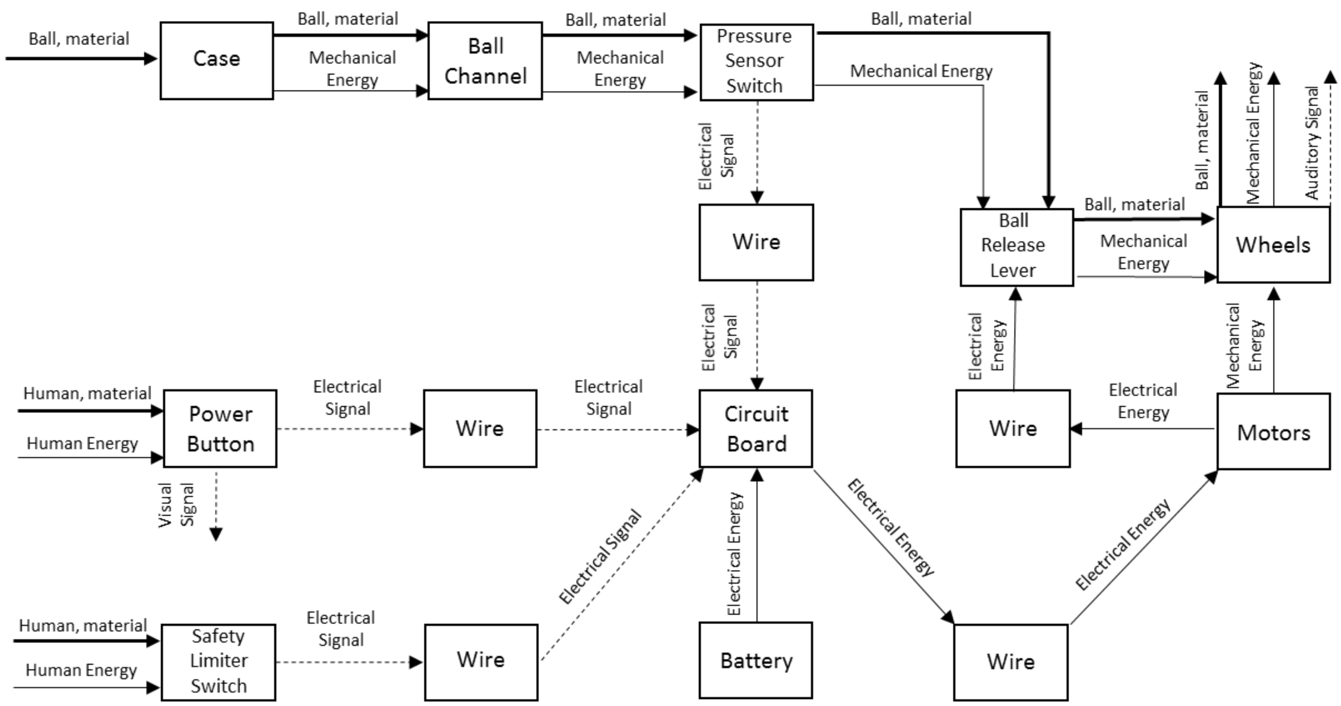

Fig. 9 iFetch Component-Flow Diagram

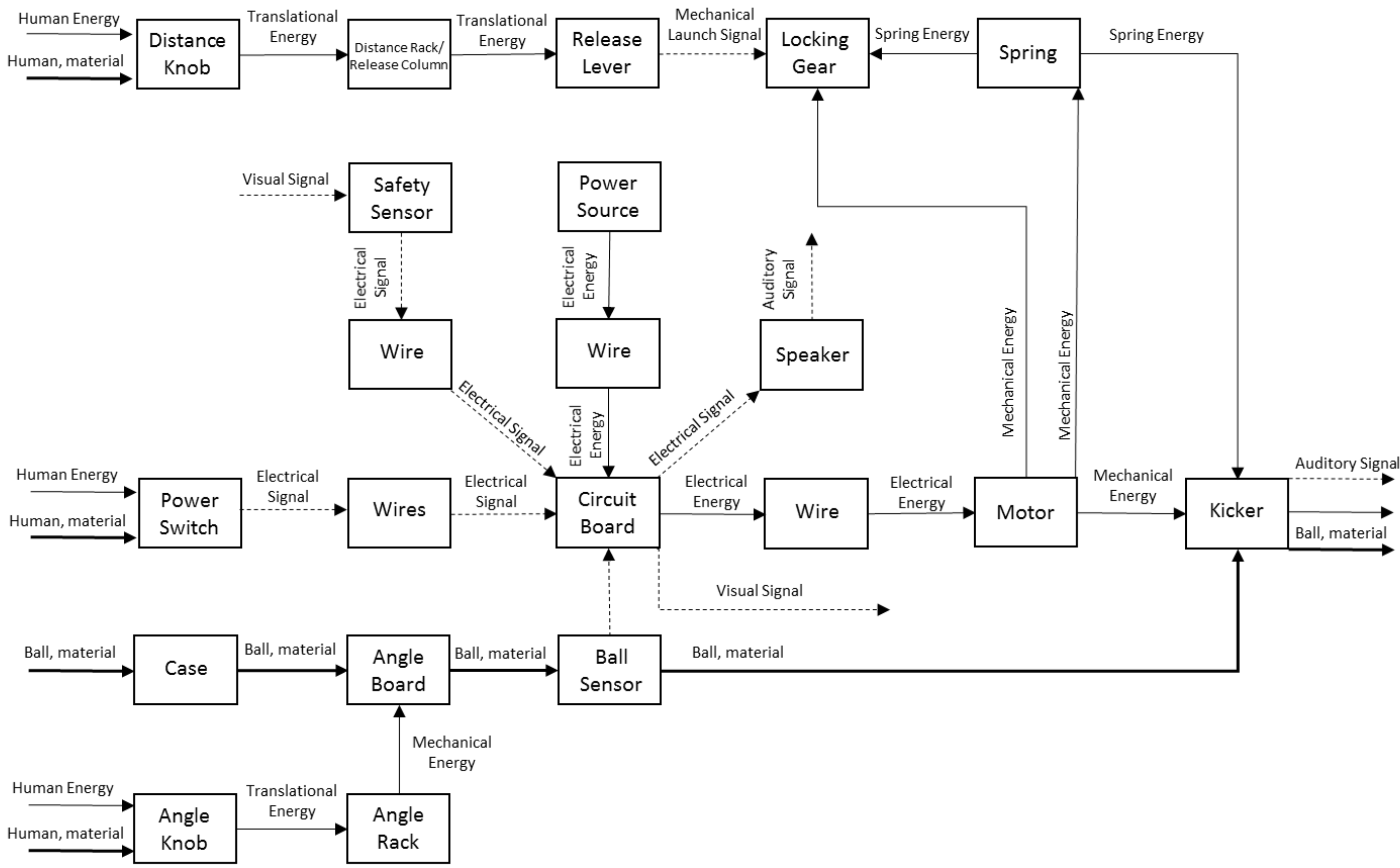

Fig. 10 PetSafe Component-Flow Diagram

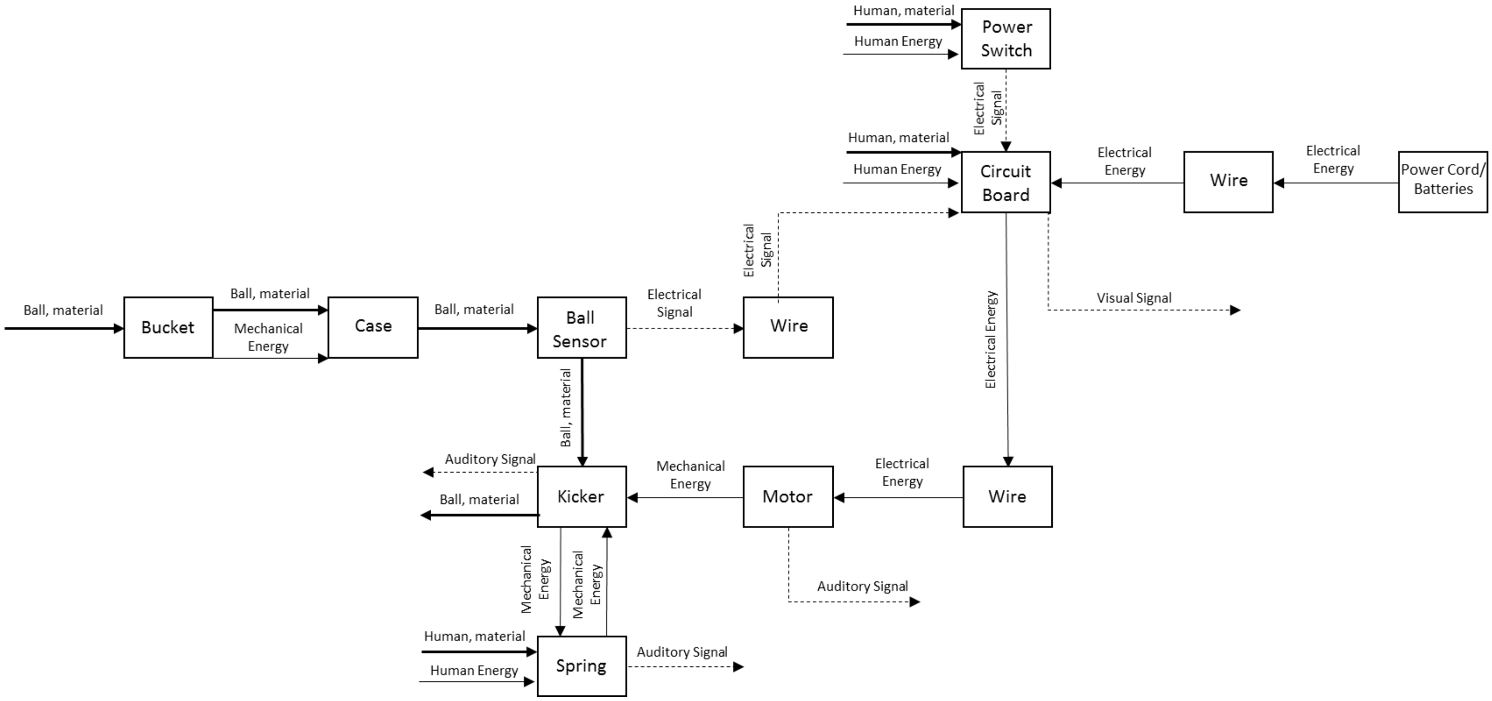

Fig. 11 GoDogGo Component-Flow Diagram