User Driven Product Innovation

by

Jayraj Nair

Master of Business Administration
Iowa State University, 1997

Bachelor of Engineering. Computer Engineering
Amravati University, 1989

SUBMITTED TO THE SYSTEM DESIGN AND MANAGEMENT PROGRAM IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN ENGINEERING AND MANAGEMENT

AT THE

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2007

Signature of Author_____

Jayraj Nair
System Design and Management Program
May, 2007

Certified by_____

James M. Utterback
David J. McGrath jr (1959) Professor of Management and Innovation
Professor of Engineering Systems
Thesis Supervisor

Accepted by_____

Pat Hale
Director, System Design and Management Program

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Accelerating diffusion of innovation to end users and enabling faster adoption is essential to product developers, especially in the industries having a rapid pace of innovation. The ability of innovators to engage with the user community to understand their needs, motivations and top issues is critical to developing products that hit the mark on meeting user needs. In the computing industry there is a need to evolve the innovation development process in parallel to the exponential growth in complexity of the products and the broad ecosystem support that is required to meet user expectations.

There are many paths to engage a user community and to obtain end user insights to create a product vision and new usage models. A simple "proof of concept" framework extending product research and development to the end user community is articulated herein. This proof of concept framework is defined in the context of a platform - a collection of ingredients that work together to meet user need. Proof of concept is conducted with users prior to general availability of a product with early ingredients that are in the research and development pipeline.

All business users of new product platforms do not adopt an innovation at the same time and can be qualitatively placed in widely accepted classification of adopter categories based on their receptivity to adopting a new product. The leading adopter categories in order are the innovator, early adopter and early majority. These categories of users may help shape current and future generations of a product specifically by validating usage scenarios with integration and deployment of a product under development in real user settings, and by helping to define trends and map requirements for future generations of platform capabilities. Using the proof of concept framework in this way helps ensure that when a product goes to market, it simply works and meets user expectations. The importance of recognizing a user need cannot be understated. The user feedback from the platform proof of concept stimulates research and development activities to address specific user needs in the current or future generations of a product platform.

There are multiple communication channels for potential adopters of an innovation. Mass media channels are effective means to create awareness of an innovation. Proof of concepts with potential users enables more rapid eventual diffusion by translating user deployment and integration learning's into product characteristics that are broadly appealing to potential adopters.

ACKNOWLEDGEMENTS

I would like to acknowledge the invaluable guidance provided by Professor James Utterback throughout the course of the thesis research. The critical analysis skills gained through the interactions with Professor Utterback in analyzing diffusion of innovations and applying the framework with potential adopters of an innovative platform vPro® from Intel Corporation was very valuable. This learning started within the first month of joining the System Design and Management Program in Spring 2005 when I had the opportunity to participate in Dr Utterback's "Disruptive Technologies-Are you Predator or Prey?" class at Sloan School of Management at MIT. Over the last two years, I have gained valuable insights through numerous class discussions with leading faculty at MIT and my fellow SDM classmates.

I would like to express my sincere appreciation to Prasad Rampalli, Upendra Kulkarni, Steve Deutsch and Sunil Kaimal at Intel Corporation for encouraging, supporting and providing me with once in a lifetime opportunity to attend the System Design and Management Program as a distance student while working full time at Intel. I am indebted to many of my colleagues, team members from the Intel® vPro™ end user initiative and friends at Intel who helped me navigate a busy academic schedule with an equally challenging and rewarding work schedule.

A special thanks to my older son Suraj, who made sure that I kept up with my academic coursework every week by checking in with the question "Are you done with your homework?" Over the two years, working our respective homework's and assignments sitting next to each other created a special bond between us. A very special note of appreciation for my mother, sister and younger son Sujay who have always been there for me and have been a tremendous source of support and encouragement during my academic endeavor at MIT.

TABLE OF CONTENTS

1. Introduction & Literature Review

1.1 Objective

Define a framework for innovation development that begins prior to the innovation diffusion process. Establish this framework for extending research and development with the user community that precedes commercial release of an innovative product. The framework should be prescriptive enough to articulate the following

Define an iterative model for a user proof of concept with tied to the innovation development process

1. Identification and selection of user to conduct the proof of concepts.
2. Implications to usage models, architecture and reference stack.
3. Capturing, translating and implementing user feedback for current & future generation of innovation.
4. Method for creating marketing and diffusion enabling collateral for potential adopters of the innovation.

1.2 Platform concepts

A degree of commoditization in the computing industry has been eroding average selling prices. The declining margins have created a shift in who is driving innovation in the industry. At Intel Corporation there has been a transition to platform focus. A platform is a set of ingredients that work together to satisfy a user need. Intel creates platforms composed of Intel ingredients and collaborates with the ecosystem partners. A platform delivers greater end-user benefit from the sum of individual parts that user's value. Intel innovates at the transistor, product and architecture level, and collectively with ecosystem partners innovates at the platform level.

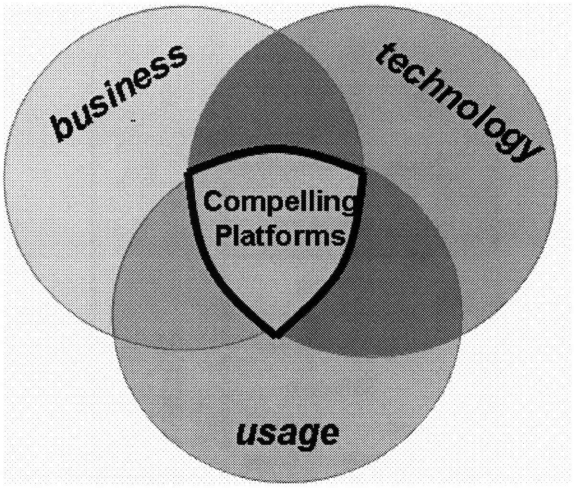

Figure 1: Compelling platforms: Source Intel Corporation

The transition to platforms helps deliver the capabilities end-users want and translates to clear value for the end-user. The focus also shifts to User driven innovation with creation and delivery of new usage models. The engagement with users in the innovation development process is to align technologies, hardware, software, the ecosystem with end-user requirements and define products for the end user. Validation of the platform solution with trusted end users against the end-user requirements, usage models would ensure that user experience is compelling.

Four levers for platform leadership as articulated in Intel's Strategic principles for Platform Leadership (Platform Leadership by Annabelle Gawer & Michael A. Cusumano HBS 2002) are the following

1. Scope of firm: What innovation development happens inside and outside?
2. Product Technology: Decisions regarding architecture, interfaces and intellectual property
3. Relationship with external complementors: Collaborative versus competitive relationships
4. Internal organization: Organization to support the above three levers.

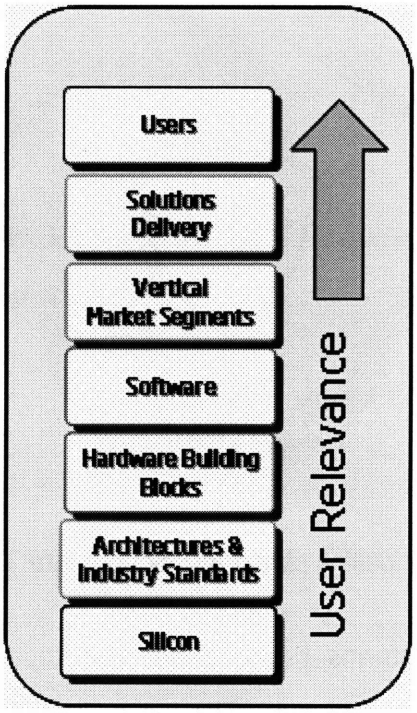

Figure 2: End user value drives platform definition: Source - Intel Corporation

An example of Intel platform is Intel® vPro™ desktop platform. It brings together the microprocessor technology, chipset ingredients on the motherboard, communications chips and relevant firmware and software from within Intel and collaborative partners in the ecosystem. The integrated ingredient stack delivers an end-user experience with differentiated value. Driving better ingredients and addressing growing architectural interdependence is achieved through platform-level planning. Platform level planning brings together all the ingredients that are necessary to deliver the user value proposition for the platform. Complexity of architectural interdependence of the ingredients is addressed by modularity. Modularity is a simple design structure in which parameters and tasks are interdependent within units (modules) and independent across them as articulated in Design Rules – The Power of Modularity (Carliss Y Baldwin and Kim B. Clark – MIT 2000). It increases the range of manageable complexity by limiting the scope of interaction between elements or tasks, allows different parts of large design to be worked on concurrently and accommodates uncertainty.

1.3 Innovation development process - Platform life cycle

Innovation is defined as an idea, practice or object that is perceived as new to an individual or another unit of adoption as articulated in Diffusion of innovations (Everett M. Rogers 2003). The innovation development process consists of all the decisions, activities and their impacts that occur from recognition of a need or problem, through research, development and commercialization of an innovation, through diffusion and adoption of the innovation by users, to its consequences.

A platform that is planned, designed, architected and engineered with user needs from the start would have a better rate of adoption and will deliver compelling user value. Technology is only relevant if it delivers compelling user experiences. User engagement upstream in the innovation process enables a user centric product definition.

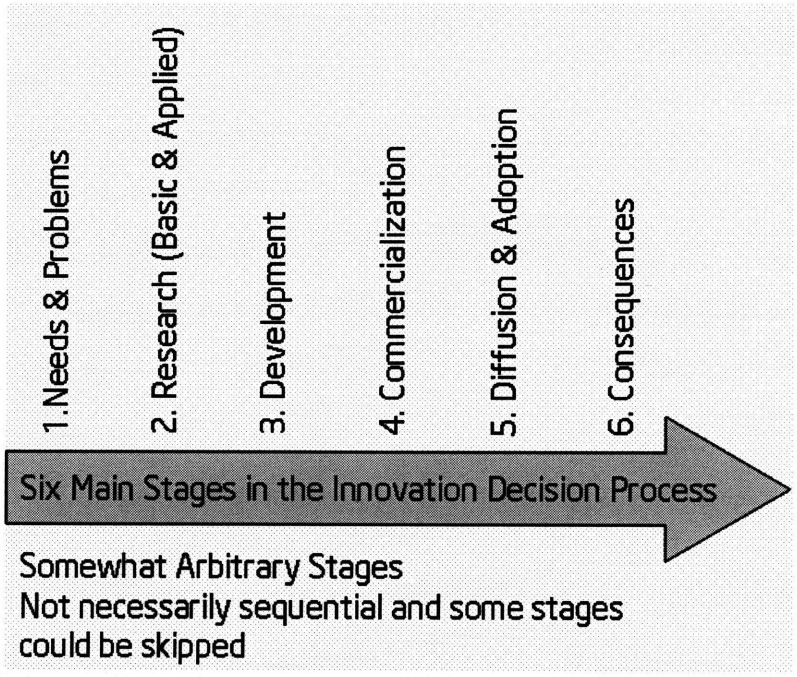

Figure 3: Six main stages in the innovation-decision process: Source - Diffusion of Innovations (Everett M. Rogers. Pg 138)

Initial stages of the innovation decision process like needs identification, problem definition, basic and applied research can be logically grouped as "Explore" and "Planning" phase of the platform life cycle. The "Development" stage can be a standalone phase. Stages of commercialization, diffusion and adoption can be logically grouped into "Deploy" phase. This logical mapping of the stages of innovation decision process to platform life cycle phases is used in subsequent chapters.

A platform life cycle starts with identification of user needs by engagement with end user driven joint research in explore and planning phase. As the platform gets developed in the development phase, early ingredients could be validated in user setting to meet their expectation. This could be achieved by means of user driven proof of concepts. The feedback and key learning's can harden current generation of products and/or help scope future generation platforms.

The diffusion enabling collateral, testimonials and adopter consequences will enable the deploy phase of the life cycle. Specific example of the "Proof of Concept" framework and its intercepts in the platform life cycle is articulated in Chapter 3. The time for a new platform to cycle from explore to deploy phase varies and can be anywhere from 12 to 36 months.

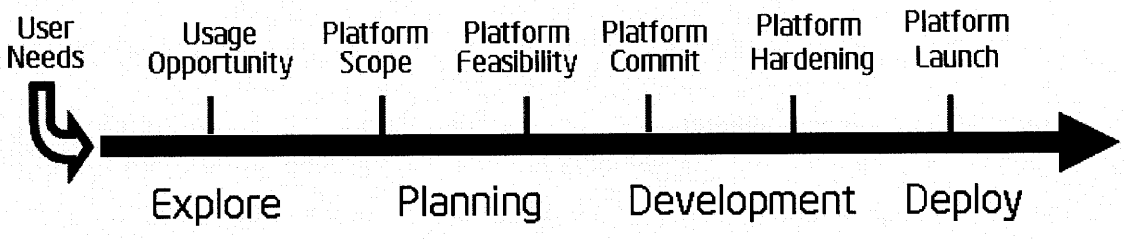

Figure 4: Platform life cycle: Source - Intel Corporation

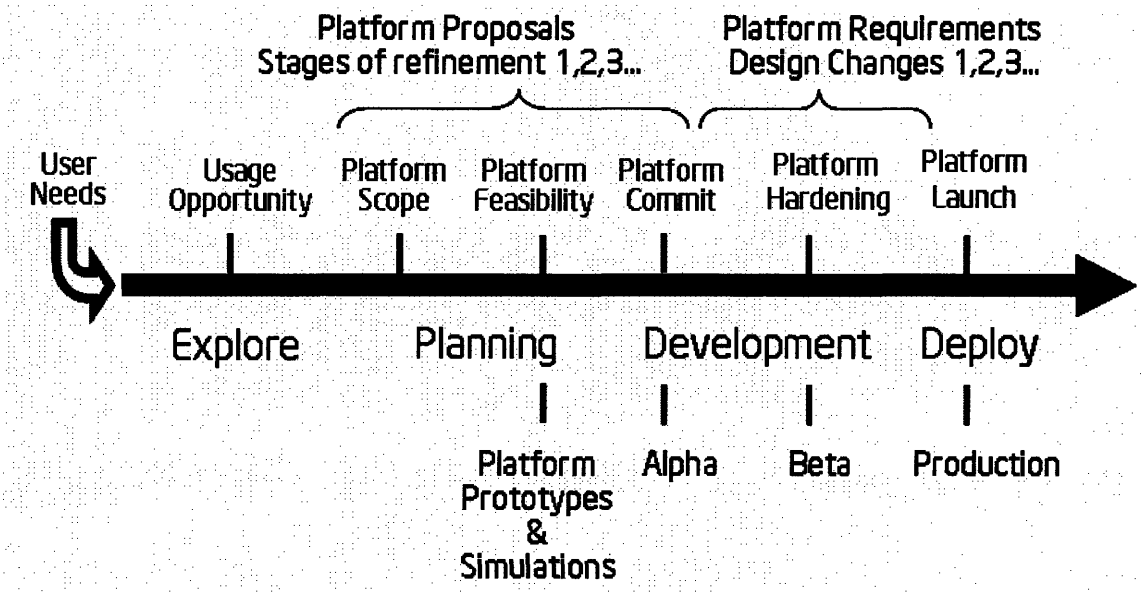

Figure 5: Platform life cycle – user engagement points with platform proposals: Source – Intel Corporation

The user engagement during explore phase translates to specific capabilities in the platform with capability/feature proposals and changes to any features and capability during the development stages are managed through platform design or requirements changes. A generic platform development sequence is highlighted to identify the availability of alpha, beta ingredients prior to the actual production release of a platform. The availability of early ingredients will become a critical element of the proof of concept framework.

1.4 Innovativeness and adopter categories

Adopter categories are a means of convenience to describe potential adopters of innovation. Users do not adopt innovations at the same time. A standard method for categorizing adopters is best described in Diffusion of Innovation (Everett M. Rogers 2003). The time element of the diffusion process enables this categorization. The adoption of an innovation usually follows a normal, bell shaped curve when plotted over time on a frequency basis.

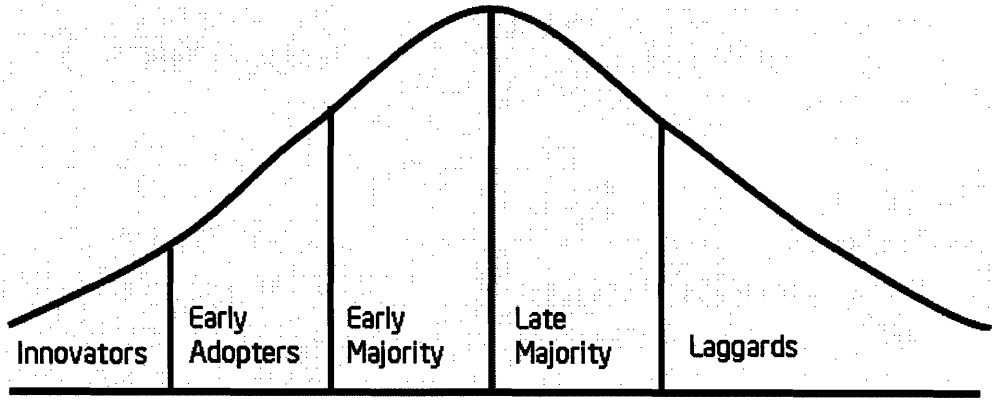

Figure 6: Adopter categorization on the basis of innovativeness: Source - Diffusion of Innovation (Everett M. Rogers 2003)

This method of adopter categorization is widely used and understood. End users of technology platforms can be logically grouped into these categories. This generalization can be applied to business users of technology. The continuum of innovativeness can be partitioned into five adopter categories (innovators, early adopters, early majority, late majority and laggards). Pronounced breaks in the innovativeness continuum do not occur between each of these categories. Innovativeness, if measured properly, is a continuous variable and there are no sharp breaks or discontinuities between adjacent adopter categories. Main characteristics and values of each adopter category are described from Diffusion of Innovation (Everett M. Rogers 2003 page 282).

Innovators: Venturesome

The salient value of the innovator is venturesomeness, due to a desire for the rash, the daring and the risky. The innovator must also be willing to accept an occasional setback when a new idea proves unsuccessful, as inevitability happens. The innovator plays an important role in the diffusion process: that of launching the new idea in the system by importing the innovation from the outside of the systems boundaries. Thus, the innovator plays a key gate keeping role in the flow of new ideas into a system.

Early Adopters: Respect

Early adopters are a more integrated part of the local social system than are innovators. This adopter category, more than any other, has the highest degree of opinion leadership in most systems. Potential adopters look to early adopters for advice and information about an innovation. The early adopter is considered by many to be "the individual to check with" before adopting a new idea. This adopter category is generally sought by change agents as a local missionary for speeding the diffusion process. Because early adopters are not too far ahead of the average individual in innovativeness, they serve as a role model for many other members of a social system. Early adopters help trigger the critical mass when they adopt an innovation.

Early majority: Deliberate

The early majority adopt new ideas just before the average member of a system. The early majority interacts frequently with their peers but seldom hold positions of opinion leadership in a system. The early majority's unique location between the very early and the relatively late to adopt makes them an important link in the diffusion process. They provide interconnectedness in the systems interpersonal networks. The early majority are one of the most numerous adopter categories, making up one third of all members of a system. Their innovation-decision period is relatively longer than that of innovators and the early adopters.

Late majority: Skeptical

The late majority adopt new ideas just after the average member of a system. Like the early majority, the late majority make up one third of the members of a system. Adoption may be both an economic necessity for the late majority and the result of increasing peer pressures. Innovations are approached with a skeptical and cautious air, and the late majority do not adopt until most others in the system have already done so.

Laggards: Traditional

Laggards are the last in a social system to adopt an innovation. They possess almost no opinion leadership. Laggards are the most localite of all adopter categories in their outlook.

1.5 Chapter summary

Platforms are complex yet modular set of ingredients that enhance user value and focus on delivering to user expectations. Intel® vPro™ desktop is an example of a platform built with a set of ingredients that is delivered by Intel Corporation and a collaborative ecosystem of partners. The development lifecycle of a platform can be represented as an iterative process of explore, planning, development and deployment. The platform launch and meeting user expectations are tightly coupled with the delivery schedules and availability of the eco-system ingredients; hence the Platform development lifecycle has to comprehend the ecosystem processes. The engagement with users starts much earlier than the deploy phase of the platform. The concepts of adopter categories and the innovativeness continuum from innovators, early adopters, early majority thru late majority and laggards are well defined by Everett M. Rogers in the Diffusion of Innovations (Fifth edition 2003) and is used as an anchor premise in understanding potential adopters.

The context, interplay between platform, the platform development life cycle and the potential adopters is at the core of this the framework for innovation development i.e. extending research and development with the user community.

Platforms are the **"what"** of the diffusion process and potential adopters - users in the respective categories are the **"who"**. The framework looks at user engagement from the innovator, early adopter and early majority categories of the system i.e. Business Users of technology. The timing and intercept of this framework to the Platform Life cycle process is critical to incorporating user input to the platform. This would constitute the **"when"** in terms of timing of user engagements.

2. Extending product research and development to the end user

2.1 End user "Proof of Concepts"

End User "Proof of Concept" is a special project with a potential adopter using the innovation and is conducted prior to the general availability of the innovation in the market place. The expected outcome of conducting an End User "Proof of Concept" is two fold:

Product Impact: Feedback that is primarily used by the developers of the innovation.

1. **Future Generation Platform Impact:** Gather trends and map/create requirements for multiyear architecture blueprint for pervasive usage models and platform features. A sample work product would be a collection of end user feedback. The feedback is assimilated, analyzed and makes its way into the product overview proposals for future platforms that are in explore and planning phase of the platform life cycle.
2. **Harden Current & Next Generation Platform:** Validate potential use cases with integration and deployment in real end user settings. A sample work product would be a list of design and requirements changes required for the platform being hardened. Based on the criticality and biz value of the required changes, these changes may be aligned with multi year platform roadmap. The impact of the hardening of the platform is significant as it detects and enables the developer to fix any critical issue prior to the general availability of the platform.

Sales & Marketing Impact: Feedback that is primarily used by the sales and marketing engine.

1. **Diffusion Enabling:** Translate end user deployment and integration learning's into case studies and whitepapers where appropriate as marketing collateral. These collateral and joint testimonials provide insight about the innovation, its applicability and value to potential adopters and can be used in available communication channels with other members of the system.

2. **Sales enabling:** Translate end user deployment and integration learning's into deployment and integration training guides such that the sales teams, who take the innovation to the user community gain a deep understanding of the real end user implementation scenarios. This toolkit of user experiences, issues and value assertions is critical insight can be taken to the user community as the product becomes generally available in the marketplace.
3. **Sales:** The "Proof of Concept" users are generally the first set of adopters that touch the innovation. As they gain the confidence of the innovation, they are also likely to purchase the innovation once it goes production and is generally available. The "Proof of Concept" partner list could be a starting seed list of potential sales.

2.2 End users – End user selection and engagement method?

The analysis and selection of key end users from the pool of potential adopters is critical to the success of the User "Proof of Concept". Revisiting the general adopter categories one would clearly want to engage with the Innovator, early adopter and early majority categories. The shortlist of users will be tied to the characteristics and usage relevance of the innovation as-well. Key guidelines for user selection is recommended

1. The product impact that the User would be able to deliver with new or incremental ideas on usage, architecture and deployment. Innovator category of potential adopter is likely to have the right users to meet this expectation. Early adopters would also have users that can generate or validate new innovative ideas.
2. Marketing and sales impact: Potential User would be able to deliver with case studies, whitepapers and diffusion enabling collateral. Early adopter category of potential adopter is likely to have the right users to meet this expectation in addition to realizing product impact. Innovators would also have users that can tell their story and learning's. Select users from the early majority category can also generate diffusion enabling collateral and testimonials.

3. It is important that the users that are selected are wiling and able to have necessary confidentiality and intellectual property agreements such that the "Proof of Concept" output can be shared with the developers and the potential adopter community at large.

4. The assessment of total available market (TAM) and market segment share (MSS) for the innovation should be considered once the above three guidelines are applied. The innovation could have a global market and the selection of users ought to deliver a global footprint such that the geography and market specific feedback for product impact, local language marketing collateral and testimonials can be obtained. Established and emerging markets for the innovation may identify differing needs and priorities for the platform.

5. Relationships matter, the ability of the "User" and the "Developer" of the innovation to partner up and work together effectively is critical. The leadership at the "User" and commitment tops down and bottoms up will yield desired results. The key is to sustain momentum and joint results in a "Proof of Concept".

It is impractical to define an algorithm to determine the number of "Proof of Concepts" that is needed to realize satisfactory product and marketing impact. One innovative user engagement can generate the "aha" idea, identify issues and help them get fixed prior to product launch or may require a handful of engagements. On the marketing and design win aspect, good coverage with case studies and key learning's that resonate with potential adopters in the system increase the likelihood of purchase. Selection of the right "Users" can accelerate the rate of adoption and translate into meaningful sales.

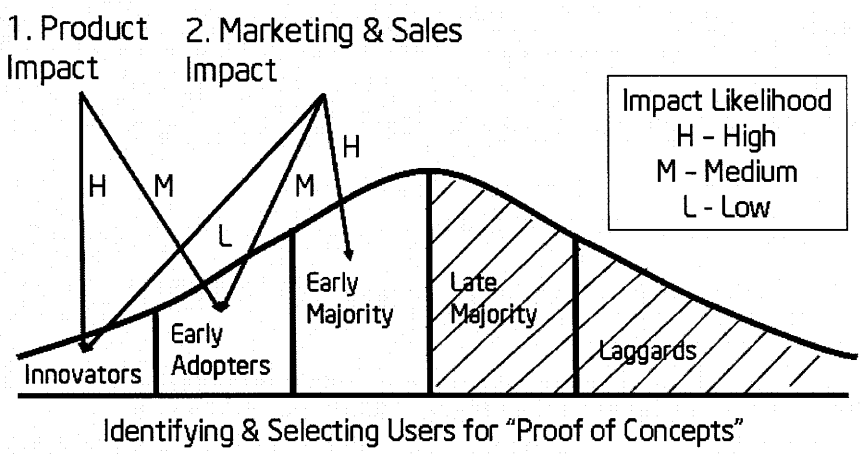

Figure 7: User "Proof of Concept" - identifying and selecting users.

Assessing the total available market (TAM) for the innovation can be achieved by market research. Users selected to provide input data have an important limitation as articulated in "Sources of Innovation" (Eric von Hippel – Oxford University Press 1988). Their insights into new products needs and potential solutions are constrained by their real world experience. Users steeped in the present are thus, unlikely to generate novel product concepts that conflict with the familiar. Although the insights of lead users are as constrained to the familiar as those of other users, lead users are familiar with conditions that lie in the future for most and are in a position to provide accurate data on needs related to such future conditions. Lead users of a novel or enhanced product display two characteristics with respect to it.

1. Lead users face needs that will be general in a marketplace, but they face them months and years before the bulk of that marketplace encounters them, and
2. Lead users are positioned to benefit significantly by obtaining a solution to those needs.

Selection of users in the potential adopter categories of innovator, early adopter and early majority needs to consider the total available market for the innovation by geography.

If the innovation is expected to resonate with users in an emerging market versus established markets then a proportional number of users "Proof of Concepts" should be conducted in emerging markets. If the identified potential adopter community breakdown is 40% in Asia, 20% in Europe, 20% in the Americas and 20% other then a similar percentage of the total number of "Proof of Concepts" should be conducted in the respective geographies. The number of "Proof of Concepts" should not be a pure numbers game and has a qualitative aspect to it, due consideration should be given to the momentum and value of the engagements.

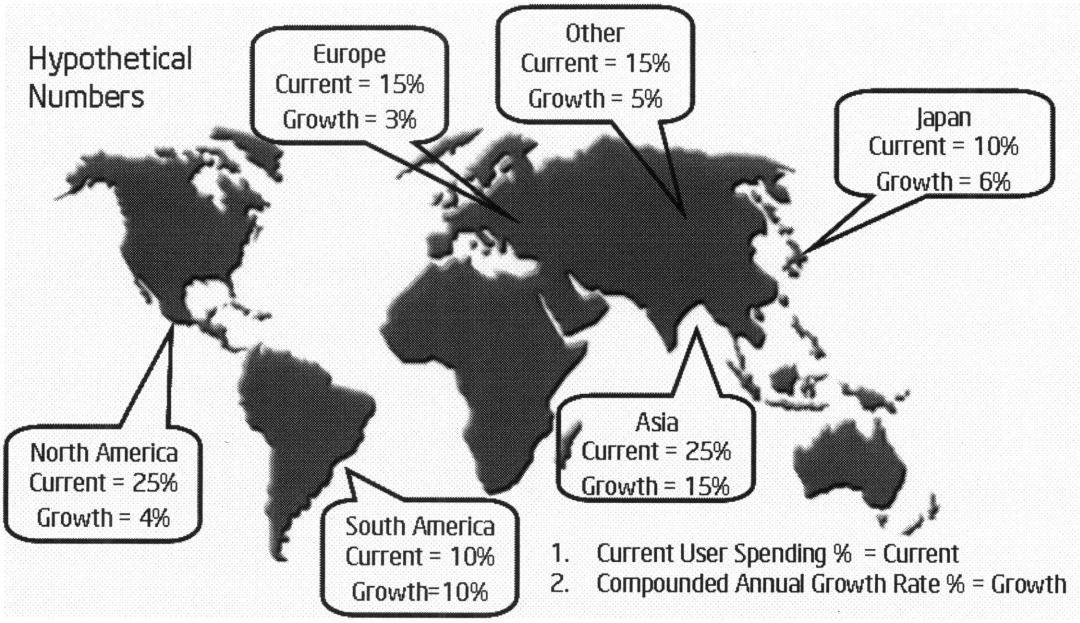

Figure 8: User "Proof of Concept" total available market assessment

A set of guidelines in priority order for identifying and selecting users follow:

Guideline-1: Users that can provide the highest product Impact - new ideas, issues etc.
Guideline-2: Users that yield marketing & sales impact - case studies, testimonials etc
Guideline-3: Users are able to work with confidentiality & intellectual property constraints
Guideline-4: Users represent their geographic markets and represent a global footprint
Guideline-5: Users have a good relationship and commitments are understood

Some or all of these guidelines can be applied in the identification and selection process. It depends on the innovation and the affinity of the potential adopter to the innovation. Many other subjective criteria's like one to many touch points with service providers can be considered to impact a broader set of potential adopters.

2.3 Model of a user "Proof of Concept"

A Proof of Concept on a new innovation, say Platform "X" requires three elements to be realize the Product and Sales Marketing Impact. They are as follows:

1. **Usage Model:** Describes overall system usage within a specific context, at a level that identifies the system's benefits to the user and includes use cases - a specific sequence of interactions between a system and one or more users including exceptions and variations. It is also beneficial to articulate the use case as a scenario, a narrative story about system use under specific conditions, generally excluding exceptions and variations.
2. **Reference Architecture:** Describes platform and integration architecture. The content is an in-depth layout of the capability and how it integrates into the user environment. The platform architecture is represented with its interfaces and exchanges in the context of how it fits with other elements of the user's environment. The technical depth and breadth of the content and its exchange with the user community should be determined based on the knowledge and level of engagement of the participants.
3. **Reference Stack:** Includes simulated and/or real hardware and software ingredients that enable the usage model and specific scenarios of relevance to the user. The "Proof of Concepts" is conducted with early ingredients i.e. alpha, beta stacks of the platform because of the timing of it being pre-production. There are inherent challenges with deploying these pre-production test platforms in user settings. They provide the first real run of the ingredients outside of the development and validation labs. Essential ingredients from the ecosystem partners that enable the usage scenarios are also part of this reference stack.

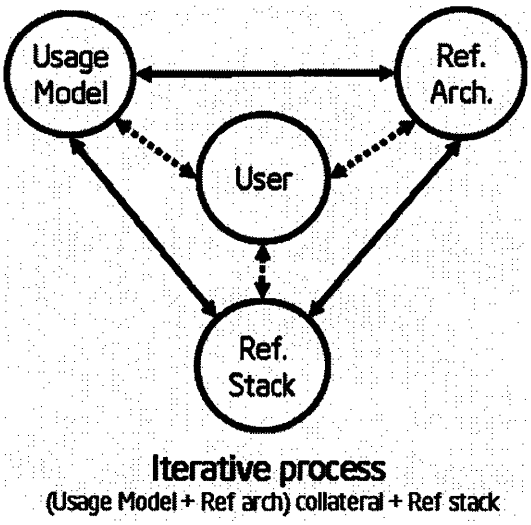

Figure 9: Three major elements of a user "Proof of Concept"

Usage Model: Describes overall system usage within a specific context, at a level that identifies the system's benefits to the user and includes use cases - a specific sequence of interactions between a system and one or more users including exceptions and variations.

Intel® vPro™ desktop platform and one of its capability i.e. built-in manageability. The usage models are defined and interactively discussed with the user in a walk-the flow setting. At the inception of the "Proof of concept" a set of identified pre-defined usage models are provided to kick start the discussions. These evolve over time and sometime result in new and incremental usage model.

Let's call the set of initial usage models as "UC Set 1.a". A typical "UC Set 1.a" would contain initial use cases, value assertions and scenarios well defined in a narrative fashion. Over the life of the project the set evolves to a higher state of clarity and quality and we can call it "UC Set 1.z". Platform inventory management to reduce or eliminate manual inventory audits by being able to locate systems regardless of its power state, health and collect inventory of the system follows.

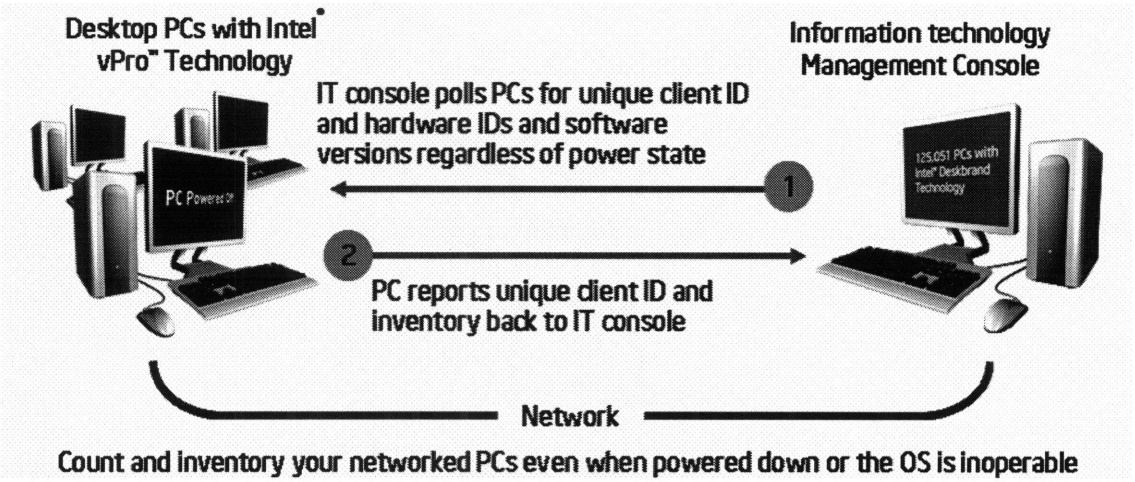

Figure 10: Sample Use Case – Remote Asset Inventory "UC Set 1.a": Built-in manageability in Intel® vPro™ desktop platform, Source: Intel Corporation

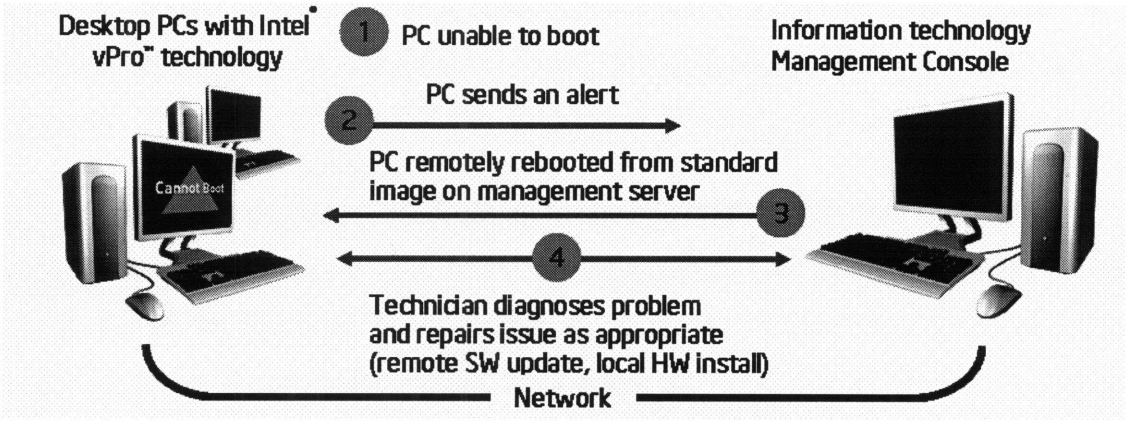

Figure 11: Sample Use Case – Remote Diagnosis and Repair "UC Set 1.a": Built-in manageability in Intel® vPro™ desktop platform, Source: Intel Corporation

The Usage Model evolves over time, as learning's and knowledge exchange progresses between the user and the developers a final version of the Usage models is articulated for "Go-to-market" communication. The key to remember is that most often the final usage model set "UC Set 1.z" is different than the starting set "UC Set 1.a". The "UC Set 1.z" also evolves to contains specific walk the flow scenario's that are easy to understand and can be broadly communicated within User "Proof of Concept" participants.

A list of key attributes to keep in mind while developing the comprehensive Use case document includes the following.

1. Use case id and title , summary of the use case
2. Goal of the use case, key actors in the use case
3. Pre-conditions & Post-conditions in the use case
4. Basic course of events - step by step narrative of action steps
5. Alternate paths to achieving the same results
6. Exceptions and extension points, business rules and opens
7. Author, Date, Version, Applicability and value assertions
8. ..... more details to describe the usage model in clear and simple terms.

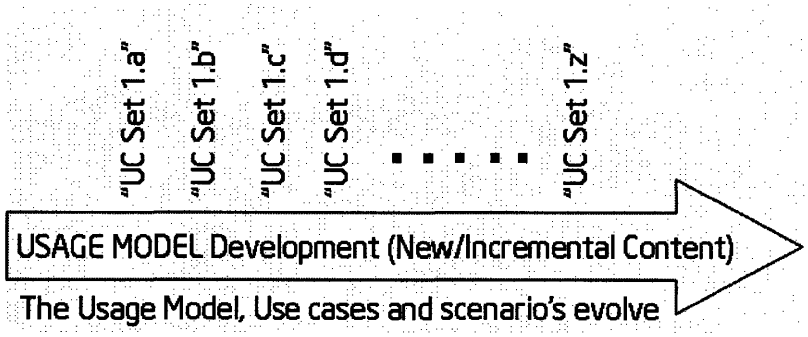

Figure 12: Usage models evolve during the user "Proof of Concept"

Reference Architecture: Describes platform and integration architecture. The content is in-depth layout of the capability and how it integrates into the user environment. The platform architecture is represented with its interfaces and exchanges in the context of how it fits with other elements in the user's environment. The starting point in a typical User "Proof of Concept" architecture engagement is a deep dive discussion about how the user capability is currently architected in the user environment. A good approach is to start out understanding how things are done today and mapping out the infrastructure ingredients associated with the usage model.

Once the underlying technology landscape is mapped out then specific macro and micro architecture design of the new capability is developed in the context of how it enhances and improves the current implementation architecture. A specific example of Intel® vPro™ desktop platform is explained to continue the linkage to the above described usage model. Establishing a good understanding of the innovation and its architecture is critical to assessing and defining the integration architecture.

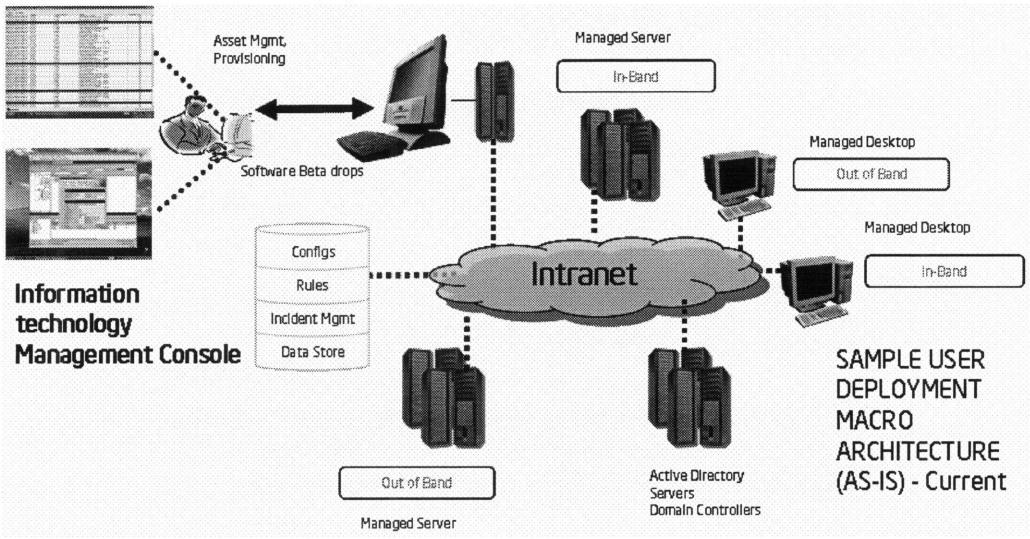

Figure 13: Sample macro architecture in user setting (AS-IS) for "Proof of Concept"

The reference architecture content evolves over time and as learning's and knowledge exchange progresses between the user and the developers, a final version of the integration architecture developed.

Often the final reference architecture set "RA Set 1.z" is different than the starting set "RA Set 1.a". The reference architecture involves a deep dive on the platform architecture and its underlying capabilities. It involves deep dives, focused exchange with technical experts to address opportunities and concerns regarding deployment architecture.

Once the current platform architecture is understood, we should consider how the new capabilities integrate and transform the environment. This could be achieved by defining a "To-Be" Architecture.

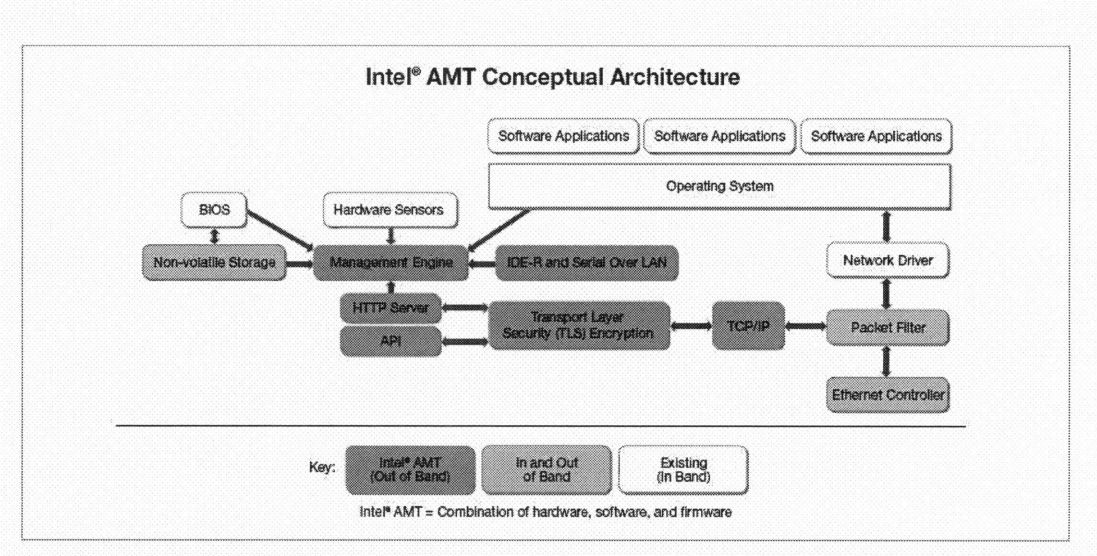

Figure 14: Conceptual architecture to enable usage models

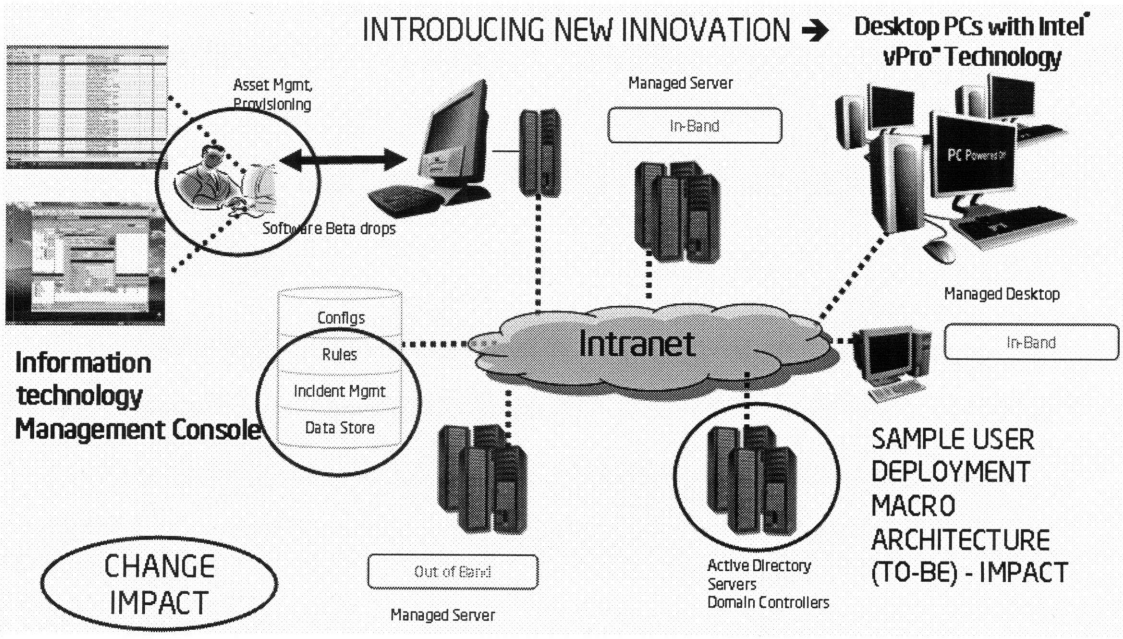

Figure 15: Sample macro architecture in user setting (TO-BE) for "Proof of Concept"

The reference architecture set can be defined as "RA Set 1.a" evolving to "RA Set 1.z" over the course of the User "Proof of Concept". The progression includes assessment of the current integration architecture, details at a macro and micro levels of the platform architecture including change impact caused by introducing the new innovation into the deployment architecture.

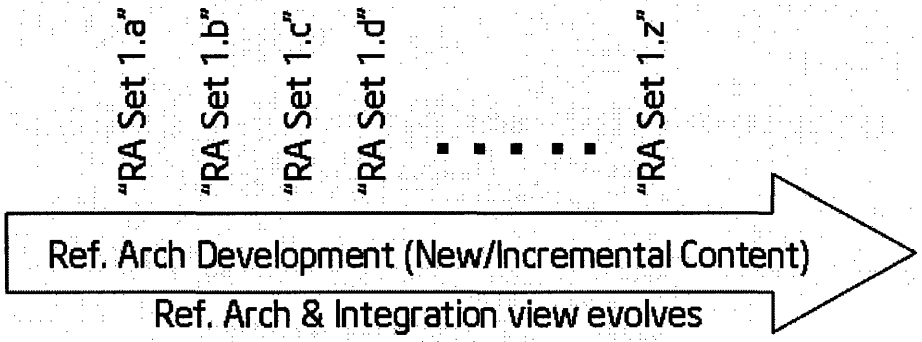

Figure 16: Reference architecture evolves during the User "Proof of Concept"

Reference stack: Includes simulated and/or real hardware and software ingredients that enable the usage model and specific scenarios of relevance to the user. The "Proof of Concepts" is conducted with early ingredients i.e. alpha, beta stacks of the platform tied to availability of pre-production. To understand the evolution of the reference stack, a macro perspective of the development lifecycle is illustrated.

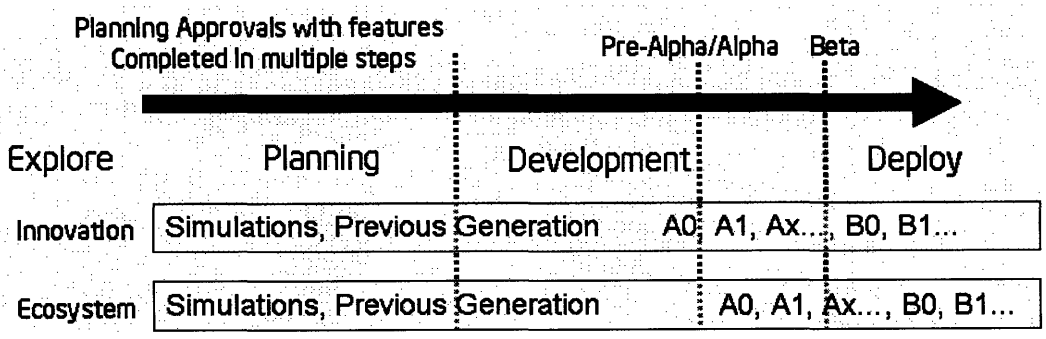

Figure 17: User "Proof of Concept" reference stack is usually pre-production

The reference stack set can be defined as "RS Set 1.a" evolving to "RS Set 1.z" over the course of the User "Proof of Concept". The progression includes evolution of the innovation as it develops along with the ingredients from the ecosystem. The reference stack needs to include all necessary ingredients to realize a specific usage model.

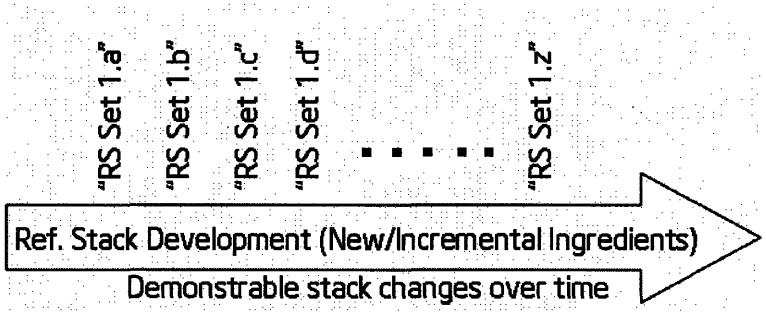

Figure 18: Reference stack evolves during the User "Proof of Concept"

The three elements usage models, reference architecture and reference stack evolve at differing schedules. The usage model is based on user requirements, needs and is the first set of content that drives the reference architecture and is inclusive of platform architecture. The reference stack is a demonstrable stack that usually follows. During the course of "Proof of Concept" the three elements evolve and change management of the package should be well understood and managed.

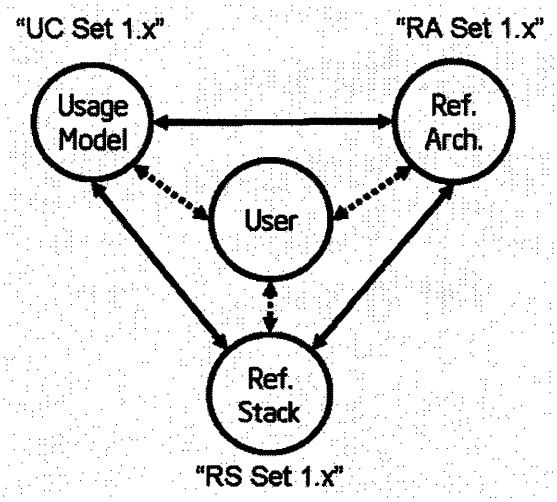

Figure 19: User "Proof of Concept" package is a managed set of three elements

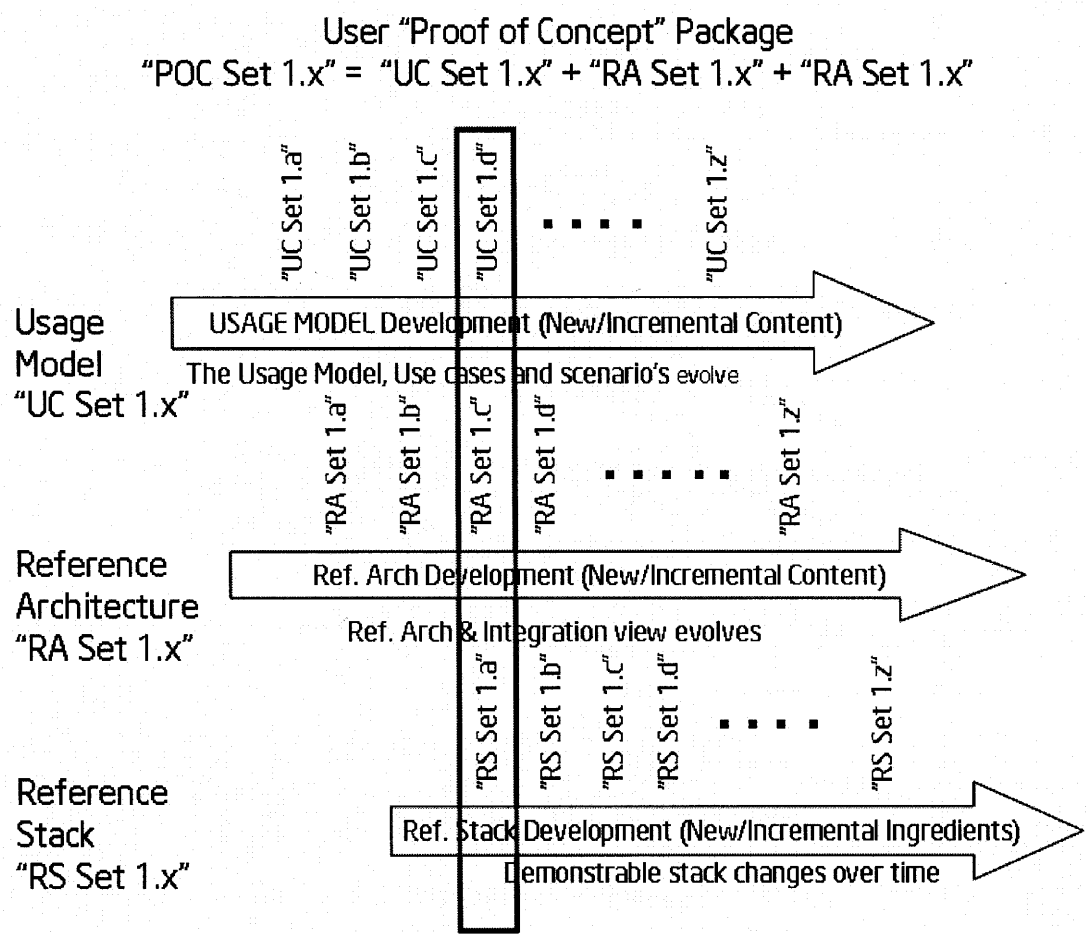

Figure 20: User "Proof of Concept" package "POC Set 1.x" is a managed set of three elements.

2.4 Types of user "Proof of Concept" and their timing

A key question is when to start these engagements. The "Proof of Concept" model iterates between usage models, reference architecture and a real reference stack so a start date for the user "Proof of Concept" is contingent on the availability of simulated or early ingredients.

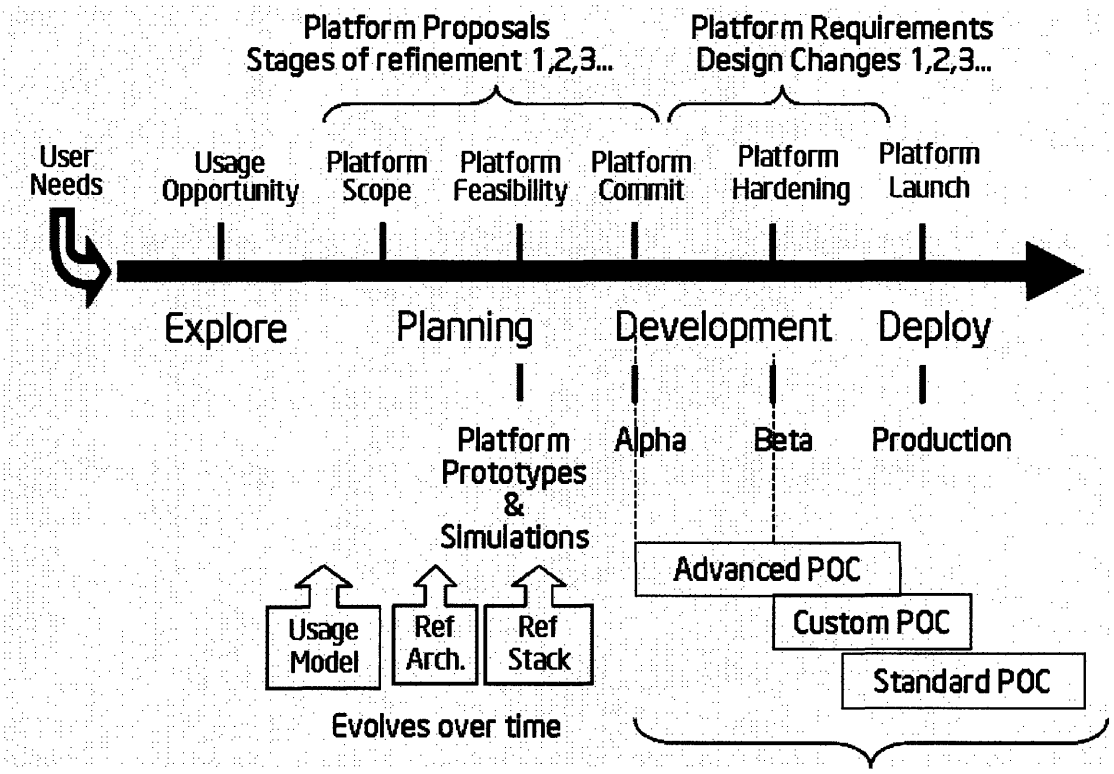

Figure 21: User "Proof of Concept" window for user engagement

Types of user "Proof of Concepts"

Advanced: These are complex and detailed user engagements typically lasting 15-30 weeks. Likely users could be from the **Innovator** category of potential adopters. The intent should be product impact by identifying key benefits and issues with the innovation. The integration and deployment feedback should be used to harden, make the current innovation better as well as gather trends and new ideas for future innovation. In an advanced "Proof of Concept", I would characterize the engagement in a driver-drafter model of role accountability with usage model, reference architecture driven by the user while the reference stack specifics are driven by the developer. Characteristics of an advanced "Proof of Concept" follow:

1. Usage Model: Initial set of usage scenarios "UC Set 1.a" with a focus on User driven innovation to evolve it to "UC Set 1.s"

2. Reference Architecture: Initial set of platform architecture and industry leading integration architecture content driven by the User "RA Set 1.a" evolves to "RA Set 1.s"
3. Reference Stack: The stack would be the first set of simulated and/or alpha/beta ingredients and could include limited number of very early eco system ingredients to demonstrate the initial set of usage models "RS Set 1.a" evolves to "RS Set 1.s"

The User "Proof of concept" package at inception of an advanced engagement can be termed "POC set 1.a" and the ability of the user and developer to advance it is paramount. It is very likely that thru the life cycle of the project change management is extremely high. The level of effort and value on part of the User and the developer of the innovation would be the highest compared to other type of "Proof of concepts".

Custom: These could be detailed user engagements typically lasting 10-15 weeks. Likely users could be from the early adopter category of potential adopters. The intent should be product impact and marketing impact by identifying key benefits, value and issues with the innovation. The integration and deployment feedback should be used to harden or make the current innovation better as well as gather trends and new ideas for future innovation. The learning's and knowledge gained should be translated to case studies, whitepapers and testimonials as diffusion enabling collateral. The usage model, reference architecture should be jointly driven by the developer and user while the reference stack specifics are driven by the developer and the supporting ecosystem. Characteristics of a Custom "Proof of Concept" follow:

1. Usage Model: Enhanced set of usage scenarios "UC Set 1.c" with a focus on User driven innovation to evolve it to "UC Set 1.s"
2. Reference Architecture: Enhanced set of platform architecture and industry leading integration architecture content driven by the user "RA Set 1.c" evolves to "RA Set 1.s"
3. Reference Stack: The stack would be further downstream set of simulated, beta ingredients, could include limited number of production or early eco system ingredients to demonstrate initial set of usage models "RS Set 1.c" evolves to "RS Set 1.s"

The user "Proof of concept" package at inception of a custom engagement can be termed "POC set 1.c" and the ability of the user and developer to integrate this configuration is important. The level of effort and value on part of the User and the developer of the innovation would be the moderate compared to other types of "Proof of concepts".

Standard: These could be simple user engagements typically lasting 5-10 weeks. Likely users could be from the potential adopter category of early majority. The intent should be generating marketing and sales impact by means of diffusion enabling collateral i.e. Case studies and testimonials. Marketing the innovation and geographical footprint to address the total available market for the innovation should be the pertinent guideline to apply.

The usage model, reference architecture and stack definitions should be driven by the developer while the deployment and integration aspects are driven by the user. Characteristics of a Standard "Proof of Concept" follow:

1. Usage Model: Predefined set of usage scenarios "UC Set 1.s"
2. Reference Architecture: Standard set of platform architecture and industry specific integration architecture content "RA Set 1.s"
3. Reference Stack: The stack would be closer to production quality ingredients and could include limited number of eco system ingredients satisfying usage models "RS Set 1.s"

The user "Proof of concept" package at inception can be termed "POC set 1.s" and the ability of the developer and user to alter or change it could be very limited. It would ensure that thru the life cycle of the project change management is light. The level of effort on part of the user and the developer of the innovation would be relatively lower compared to other types of "Proof of concepts".

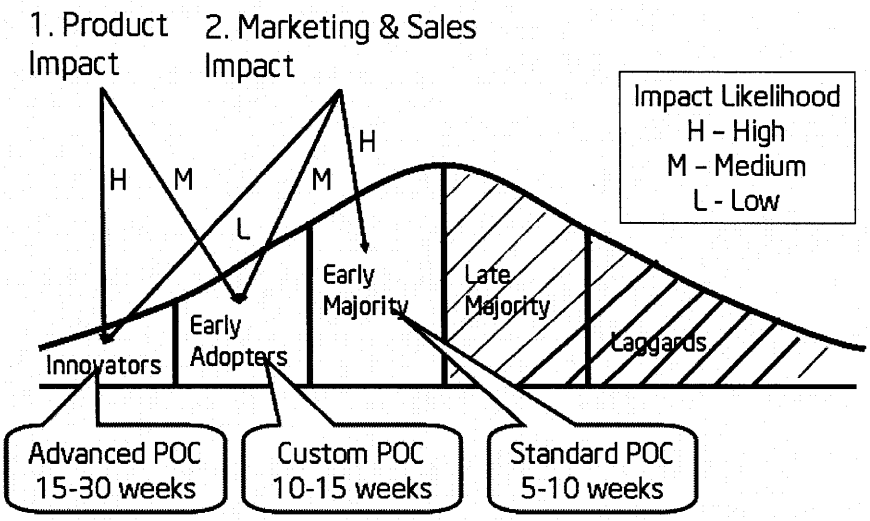

Figure 22: Mapping adopter categories to user "Proof of Concepts"

User "Proof of Concept" timeline

It is challenging to work with early ingredients and we should tie the "Proof of Concept" deliverables to the launch window due to the inherent nature of User driven innovation. Launch event for the innovation is usually a good target for containing these projects from becoming long term runaway research projects.

Similar approach with varying timelines can be applied to different types of "Proof of Concepts". In case of standard "Proof of Concepts" there would be critical milestones reflecting the case study development tasks while in the advanced "Proof of Concepts" there will a lot more emphasis on the collection, analysis of feedback for changes or innovation in the product. A potential time bound framework for the top ten task categories follows:

Milestones for Major Task Categories of User "Proof of Concept"	Weeks
Initial User & Developer engagement - Deep Dive discussions (round-1)	"t"-6
Deep dives on Usage Models, Reference Architecture and Reference Stack	"t"-4
Delivery of Reference Stack (Simulated and early ingredients) "RS Set 1.x"	"t"-2
Joint Project team in place - User and Developer of Innovation	"t"
Deployment and integration in User setting of "POC Set 1.x"	"t"+2
Analysis and Usage model evolution - mindshare and innovation	"t"+5
Midpoint review (Usage Model ⇔ ref arch ⇔ ref stack)	"t"+8
Ref Stack upgrade and final integration (25-50-75% enhancements)	"t"+12
Findings and analysis report out (Joint User and developer)	"t"+15
Publication of results - product impact findings and diffusion enabling collateral	"t"+20

t → Point in time when joint teams
Commitment is in place between User & Developer of innovation

Figure 23: Time bound approach to managing user "Proof of Concepts"

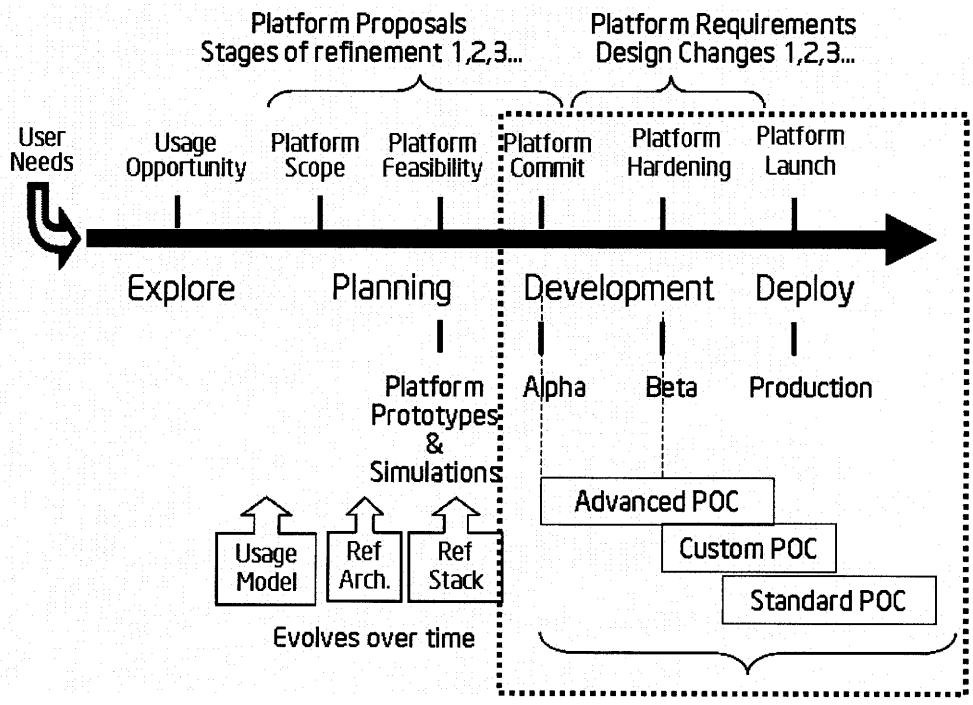

Figure 24: Link user "Proof of Concepts" of an innovation to its launch event

2.5 Chapter summary

End user "Proof of Concept" is a special project with a potential adopter using the innovation and is conducted prior to the general availability of the innovation in the market place. The expected outcome of conducting an end user "Proof of Concept" is Product Impact - feedback that is primarily used by the developers of the innovation for improving, innovating the product and sales/marketing impact – feedback that is primarily used by the sales and marketing engine for generating diffusion enabling collateral.

The identification and selection of users for the "Proof of Concept" is done from the adopter categories of Innovators, early adopters and early majority. An assessment of the total available market is equally important to ensure global footprint. Legal and intellectual property management agreements should be clear and well understood. Relationships matter and mutual expectations on the "Proof of Concept" between user and developer of the innovation must be well defined.

The product impact can be achieved by running advanced "Proof of Concept" with innovators while marketing and sales impact is primarily achieved with early majority of adopters. Early adopter can provide both product and marketing impact. The "Proof of Concepts" can be classified into three major categories Advanced, Custom and Standard "Proof of Concepts" with a run time of 15-30, 10-15 and 5-10 weeks respectively. Every "Proof of Concept" has three elements usage models, reference architecture and reference stack that evolve at differing schedules.

The usage model is based on user requirements, needs and is the first set of content that drives the Reference Architecture. The reference stack is a demonstrable stack that usually follows. During the course of "Proof of Concept" the three elements evolve and change management of the package should be well understood and managed.

It is challenging to work with early ingredients because of stability, configuration unknowns and the project should be tied to the launch of the innovation. Launch event for the innovation is usually a good target for containing these projects from becoming a long term runaway research projects. Key is to tie all activities to the launch date of the innovation i.e. when the innovation is available to all potential adopters.

A model of t+"x" weeks can be defined to manage the work items of the project. In case of Standard "Proof of Concepts" there would be critical milestones reflecting the case study development tasks while in the advanced "Proof of Concepts" there will a lot more emphasis on the collection, analysis of feedback relative to changes and innovation in the product.

It is impractical to define an algorithm to determine the number of "Proof of Concepts" that is needed to realize satisfactory product and marketing impact. One innovative user engagement can generate the "aha" idea, identify issues and help them get fixed prior to product launch or may require a handful of engagements.

On the marketing and sales aspect, good coverage with case studies and key learning's that resonate with potential adopters in the system increase the likelihood of purchase. Selection of the right "Users" and conducting the user "Proof of Concept" can accelerate the rate of adoption and translate the engagements into meaningful sales.

3. Capturing, translating and implementing user feedback

3.1 Identifying trends, needs and implementing product feedback

In an advanced or custom "Proof of Concept" a package "POC Set 1.x" is setup in the user environment. It includes the three elements, a set of usage models with use case scenarios "UC Set 1.x", architectural collateral with specifics on integration and deployment "RA Set 1.x" and the reference stack with early ingredients "RA Set 1.x".

Deep dives on capabilities of the innovation precede the deployment and integration of the innovation in a user environment. The deep dive sessions should include participation from key users and developers of the innovation. The starting point of these deep dives is a formulation of the capability vision and the underlying needs and desires of the user. User feedback from these deep dives can be categorized at a macro and micro level:

1. Macro level: Problem statements and user needs at a macro level. It would include their top issues and desires for a capability in a specific domain. Example: I don't have visibility to all the computing assets in my company and can't get accurate inventory. Can you build a capability that gives me all this information and puts it at my fingertips irrespective of time of day and by the way - all these systems that my employees power off and leave, can you give me the capability to inventory them without having to send somebody to physically locate them across my offices around the world?
2. Micro level: Specific needs from a usability, deployment and integration perspective at a micro level. It would include specifics of the capability and its functional characteristics and interoperability, integration into the user's environment. Ex: We use Microsoft domain controllers and all the security and rights management is done thru active directory, how does this innovation integrate with my existing infrastructure. What interfaces and capabilities can make your innovation deployable in my settings?

Translating user needs into platform specific features and capabilities is critical. An example of a user need expressed at a macro level in business and Information technology terms:

1. User Need expressed in business terms: Need to lower my total cost of ownership (TCO) of my desktop infrastructure, my operations costs are high and all of my distributed assets are difficult to manage.
2. User Need expressed in Information Technology terms: I need to be able to inventory my entire desktop infrastructure irrespective of their power state and remotely manage them over the wire instead of sending technicians to repair them.

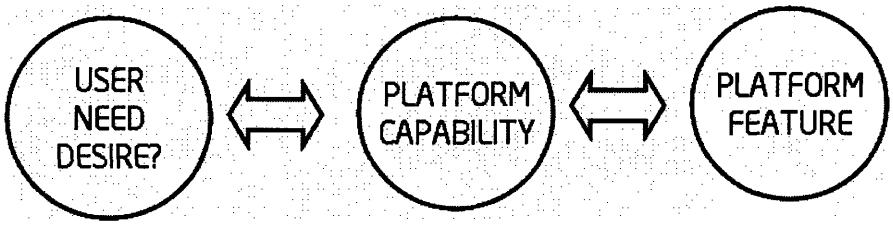

Figure 25: User needs translate to platform capability and features

User need expressed in business and information technology terms above are translated to corresponding capabilities in the desktop platform. Needs expressed above translate to a manageability capability on the desktop platform that can reduce or eliminate manual inventory audits by being able to locate systems over the wire regardless of power state or health, Improve version control and configurations inventory, reduce manual audits and better manage hardware inventory. Reduce desk side visits to resolve software problems, even when operating system is down; reduce visits to resolve hardware problems with improved remote diagnosis and hardware information.

Further translated to platform specific feature it translates to having a manageability engine on the platform with its own power system that can operate when the system is down, that can talk over the wire to a management console and can facilitate remote troubleshooting, diagnosis and repair. This subsystem needs to be able to store specific configuration of the system, inventory about it in some form of storage that is not impacted by the operating system not functioning. This manageability engine capability can potentially translate to specific features on the silicon chips on the platform and/or in firmware/software that is resident in the platform.

To realize this capability the innovator can build new features in the platform and collaborate with the ecosystem to deliver solutions that deliver to the specific user need. Not all needs can be translated and delivered in one cycle of platform development. The exercise of figuring out market viability and technical feasibility of the capability and associated feature set should be part of the planning process. The planning process needs to be agile and deliberate enough to assess the user need in the context of larger trends in the potential adopter community.

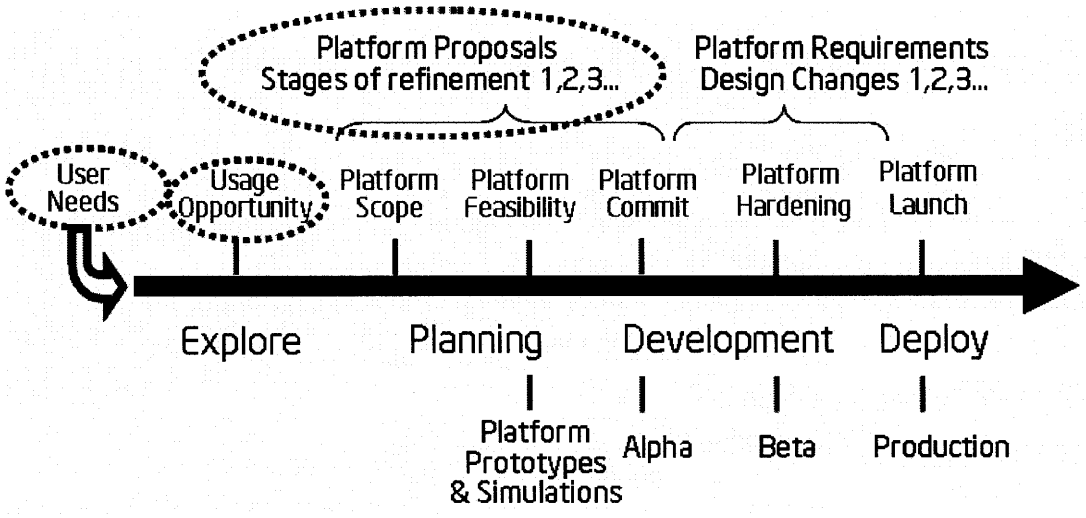

Figure 26: User needs translate to platform capability in the planning cycle

3.1 Identifying problems and product hardening

During the deep dives and "Proof of Concept" issues may surface with the innovation. In cases where some of these identified issues are significant and can impact the adoption of the innovation positively or negatively then it should be immediately fed into the planning and development cycles. Once the innovation is in the development phase, changes driven by the user feedback for the current generation of the product should be managed with discipline of change management.

A requirement change required process (RCR) should be used to capture, track and manage changes to the requirements. This would a variance from the committed product requirements that is driving the development of the innovation. The developer of the innovation should assess the impact of the change to development activities. If the issues identified during the course of the user "Proof of Concept" is critical in terms of adoption, identifying and resolving it is a big win for the developer.

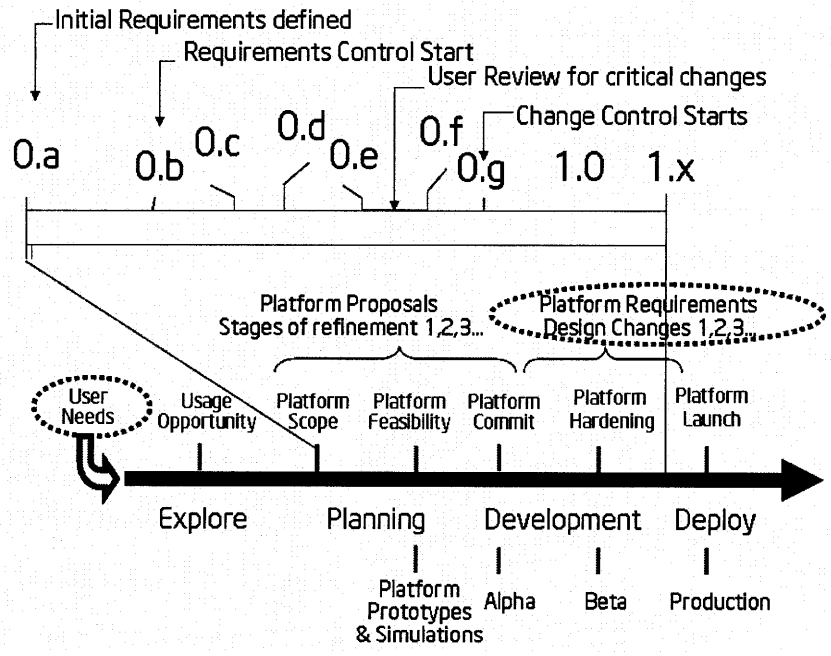

Figure 27: User needs may translate to requirements change in current innovation.

The ability and desire on part of the developer to seek critical feedback on the current innovation, resolve to change the innovation and make it better ensures a better rate of adoption. Finding adoption critical issues with the innovation and fixing them prior to general availability to potential adopters would be extremely beneficial and can be articulated as the product hardening impact.

3.3 Creating marketing and diffusion enabling collateral

The potential adopter category that can make the most impact in creating diffusion enabling collateral and testimonials is the early adopter category. This adopter category, more than any other, has the highest degree of opinion leadership in most systems. Potential adopters look to early adopters for advice and information about an innovation. The early adopter is considered by many to be "the individual to check with" before adopting a new idea. As articulated in Diffusion of Innovations (Everett M. Rogers 2003) this adopter category is generally sought by change agents as a local missionary for speeding the diffusion process. Because early adopters are not too far ahead of the average individual in innovativeness, they serve as a role model for many other members of a social system. Early adopters help trigger the critical mass when they adopt an innovation. Following very closely the early majority adopt new ideas just before the average member of a system. The early majority's unique location between the very early and the relatively late to adopt makes them an important link in the diffusion process.

An outcome of the User "Proof of Concept" is collateral that contains key learning's, benefits, value assertions and capability assessment from a user's perspective. This collateral can be packaged in the form of whitepapers, evaluation briefs and testimonials that can be communicated through different channels to potential adopters. The opinion leadership of the early adopters is very valuable in the diffusion process and provides answers to questions from many potential adopters of the innovation. The collateral can be structured in many ways depending on the communication channel.

Case studies, whitepapers and innovation evaluation briefs

Print and electronic media collateral like whitepapers and evaluation briefs should contain the following categories of information.

1. **User:** Specifics about the "Proof of Concept" user – Sets the context of the adopter and helps determine affinity for potential adopters by type, location, nature and size of the business. Example: If the "Proof of Concept" user is a large distributed manufacturing enterprise, potential adopters that belong in the specific industry category are likely to associate themselves with the collateral.
2. **Challenges and opportunities:** Sets the context of the problem statement and ties the outcome to the very specific "User Need" that led to the "Proof of Concept".
3. **Specifics on innovation being evaluated** – Maps the innovation and its capabilities to the challenges and opportunities. User needs map to capabilities of the innovation.
4. **Results and bottom-line Impact** – Articulates the findings and its impact to addressing the "User Need". This ought to be the core of the collateral and should lay out the three elements of the "Proof of Concept" i.e. Usage Models, Reference Architecture and Reference stack information. Specifics on "How" it was done would be valuable as-well.
5. **Next steps and contacts for more information** – Sets the tone on what the "Proof of Concept" user is going to do with the innovation going forward and shows their ability to share the insights and benefits of the innovation and willingness to provide a communication link to the rest of the potential adopter community.

Case study, whitepapers and evaluation briefs from early adopters and early majority users that participate in the "Proof of Concepts" takes time and effort to prepare. Key legal aspects of disclosure, copyright and brand ownership should be addressed prior to release of collateral. The collateral can contain simple messaging in sheer qualitative terms or could be data intensive, quantitative in nature regarding the value and benefit assertion. An attempt should be made to balance the quantity and quality of the collateral with respect to qualitative and quantitative content.

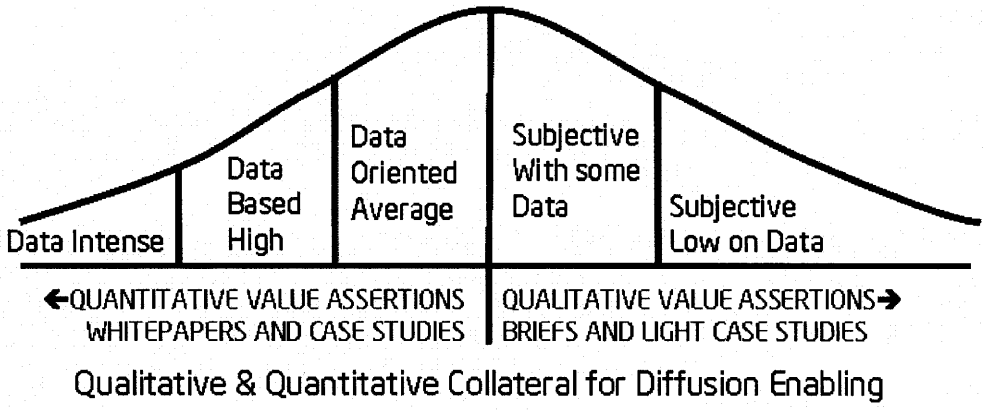

Figure 28: Types of collateral – quantitative and qualitative collateral

I have not conducted quantitative study on what type of collateral best resonates with specific categories of potential adopters but would recommend that a good mix of 50-50 quantitative-qualitative type of collateral be generated. In terms of communicating and delivering the collateral, while all the collateral can be made available broadly, specific targeting can be done by affinity of user to a comparable user by industry, size and type of business, geography and locality. Additionally an approach to consider would be to map collateral type to adopter category. Availability of all types of collateral should not be limited in any way to potential adopters but can be targeted as such.

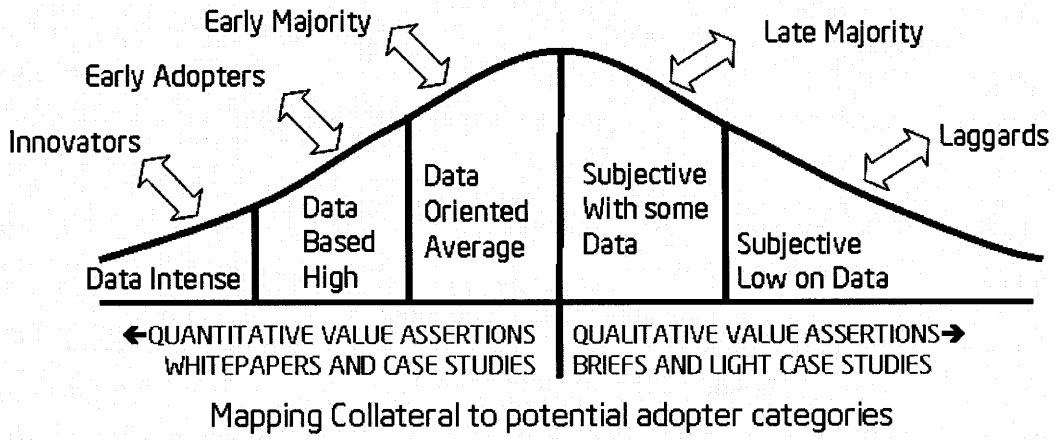

Figure 29: Targeting type of diffusion enabling collateral to adopter categories

Testimonials and joint messaging

In relevant user forums like round tables, developer forums, launch events, this collateral can be diffused to potential adopters. Joint messaging and communication from the developer and user is extremely beneficial. Value assertions and messaging coming from the user resonates best with their peers in the potential adopter community. Innovators and early adopter category of users can influence and accelerate the formation of critical mass of adopters from the early majority category.

3.4 Sustaining the feedback loop, impact to the platform product life cycle

Engagement with lead users for feedback and identification of their problems and needs should be a continuous exercise. It is not a one time discrete activity for a new innovation. There should be sustainable engagement model to get the thought leaders and innovators involved in the explore phase of current and future innovations.

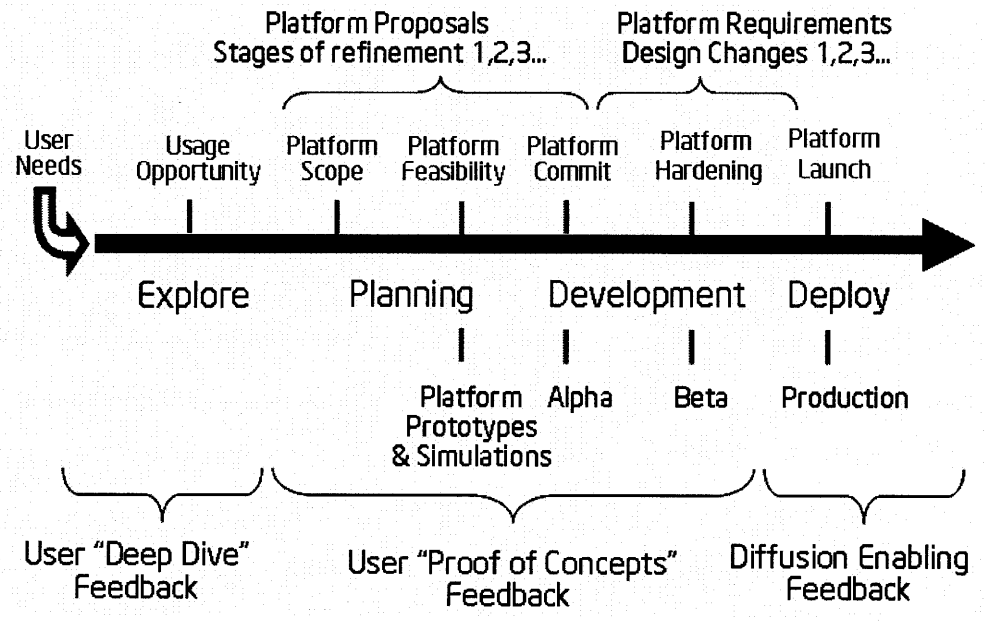

Figure 30: "Proof of Concepts" feedback impact current and future innovation.

Users who participate in the "Proof of Concept" - innovators, early majority and early majority category of potential adopters generate feedback that provides needs, trends and direction for current and future innovation. Users who participate in "proof of Concepts" expect to see their feedback incorporated in the innovation. A method to analyzing, prioritizing and incorporating the feedback in the planning and development phase of the information is required. Translating the volumes of data collected to information and mining the information for key nuggets "Aha!" is key to delivering value.

3.5 Chapter summary

User need expressed in business and Information Technology terms should be translated to corresponding capabilities in the desktop platforms. The "Proof of Concepts" identifies trends and user needs that provide direction to the developer. This insight and feedback can translate to capabilities and features in current, future generation of the innovation. The hardening of the current generation of the innovation can be achieved by fixing issues identified by the user. A requirement change required process should be used to capture, track and manage changes to the requirements. This would a variance from the committed product requirements that is driving the development of the innovation. The developer of the innovation should assess the impact of the change to development activities. If the issues identified during the course of the user "Proof of Concept" critical in terms of adoption, identifying and resolving it is a big win for the developer.

The adopter category that can make the most impact in creating diffusion enabling collateral and testimonials is the early adopter category. This adopter category, more than any other, has the highest degree of opinion leadership in most systems. Potential adopters look to early adopters for advice and information about an innovation. The early adopter is considered by many to be "the individual to check with" before adopting a new idea. Joint messaging and communication from the developer and user is extremely beneficial. Value assertions and messaging coming from the User resonates the best with their peers in the potential adopter community.

4. Conclusions

4.1 Framework and its applicability/usefulness

User "Proof of Concept" framework can be generally applied to any innovation. Working with lead users and getting them involved in the innovation process is beneficial to the user and the developer of the innovation. User "Proof of Concept" provides the opportunity to address user problems by incorporating their requirements in current and future generation of the innovation. The rate of adoption of innovation can be accelerated by working with the innovator, early adopter and early majority category of potential adopters.

The user "Proof of Concept" - simple framework of bringing together usage models, architecture and the innovation into the user environment prior to general availability is often the first trial for the innovation in real end user setting. The ability of key users in the innovator and early adopter category to impact the innovation by providing critical feedback and creating diffusion enabling collateral for the potential adopters at large is essential to the success of the framework.

The essential elements of the user "Proof of Concept" evolve over time and the expectation is that the starting set with usage models, reference architecture and reference stack is not where it ends. Early involvement from users changes the dynamics of innovation development, it is not just a push of innovation and reflects involvement and active user partnership in the innovation development process.

4.2 Challenges with the framework

The identification, selection of key users for "Proof of Concept" is critical to the framework. The users, number and type of engagements - standard, custom and advanced should be determined after analyzing key market research data.

Potential adopter total available market and geographic distribution of potential adopters for the innovation should be considered before selecting key users for the "Proof of Concept". Key challenges with the implementation of the user "Proof of Concept" can be overcome with due diligence and quality pre-work:

1. Defining the "Proof of Concept" with appropriate intellectual property and non disclosure agreements between the user and the developer of the innovation. The relationship and commitment between the developer and user of the innovation can get tested while defining a statement of work for the "Proof of Concept".

2. Setting up mutual expectations and managing change through the life cycle of the "Proof of Concept". Most engagements can start of with high enthusiasm that can decline over time as rigor of planned against schedule on deliverables is applied. The ability of the user and developer to resource and support the engagement is important as these strategic projects can sometime get relegated to long term research projects.

3. Translating critical user feedback into changes to the current generation of the innovation can be challenging as the development process is usually on a roll. As critical feedback flows into the developer and translates into required changes, the engineering change impact to the current launch and delivery schedules should be assessed in light of delivering compelling user experience at launch. Striking a good balance on prioritizing critical issues and resolving it prior to general availability is essential to the success of the "Proof of Concepts". Representative user requirements that are not reflected in the current generation of the innovation should be incorporated in the appropriate roadmaps, explore and planning cycle of subsequent generation of the innovation.

4.3 Benefits and value of this framework

A user driven innovation will deliver the most compelling user experience as it reflects the needs and requirements of the user. Partnering up with the innovators and early adopters during the research and development stages of the innovation development process can ensure that the products will meet expectations of the user. Users of the "Proof of Concept" can deliver diffusion enabling collateral that resonate with potential adopters of the innovation.

User "proof of concept" can deliver future generation platform impact by gathering trends and creating requirements for multiyear architecture blueprint, platform features led by user driven pervasive usage models. Harden current generation platform by validating potential use cases with integration and deployment in real end user settings. The impact of platform hardening is significant as it detects and enables the developer to fix any critical issue prior to the general availability of the platform.

Packaging diffusion enabling collateral with deployment and integration learning's into case studies and whitepapers is very valuable. The collateral and joint testimonials will provide insight about the innovation, its applicability and value to potential adopters and can be used in available communication channels with other members of the system. The deployment and integration learning's can be assimilated into deployment and integration training guides such that the sales teams who take the innovation to the user community gain a deep understanding of the real end user implementation scenarios.

The toolkit of user experiences, issues and value assertions is critical insight that the sales teams can take to the user community. The "Proof of Concept" users are generally the first set of adopters that touch the innovation. As they gain the confidence of the innovation, they are also likely to purchase the innovation once it goes production. The "Proof of Concept" partner list could easily translate to sales opportunity with early adopter and early majority of potential adopters.

4.4 Future work and research

The identification and selection of user for the "Proof of Concept" is a subjective method and the logical grouping of target users into potential adopter categories is not really an exact science. This aspect of user selection for the "Proof of Concepts" can benefit from further research. A model for quantifying the rate of adoption of innovation with potential adopter categories and the impact of the User "Proof of concepts" in terms of accelerating the rate of diffusion is another research area of interest.

4.5 Chapter summary

Accelerating rate of diffusion of technology Innovation to End Users and enabling faster adoption is essential to product developers and user "proof of concepts" is a good framework to bring the "User" into the innovation development process. There are many paths to engaging with the user community and leveraging end user insights to create a vision and new usage models. A simple framework to extending product research and development to the end user community will realize user driven product innovation. All business users of new platforms do not adopt innovation at the same time and can be qualitatively placed in widely accepted classification of adopter categories based on their innovativeness.

Engaging on platform "Proof of Concept" with select users in the innovator, early adopter and early majority categories will provide an impact to current and future generations of the product specifically by validating usage scenarios with integration and deployment in real user settings, gathering trends and mapping requirements to future generation of platform capabilities. It ensures that when the product goes to market, it simply works and meets user expectations. The user feedback from the platform proof of concept stimulates research and development activities to address user need in the current or future generation of the platform. The platform proof of concept enables diffusion by translating user deployment and integration learning's into case studies and whitepapers that is broadly available to potential adopters.

5. Bibliography

1. Annabelle Gawer & Michael A. Cusumano : Platform Leadership How Intel, Microsoft and Cisco drive Industry Innovation, Harvard Business School Press 2002
2. Carliss Y. Baldwin and Kim B. Clark : Design Rules The Power of Modularity, MIT 2000
3. Clayton M. Christensen : The Innovators Dilemma, Harvard Business School Press 1997
4. Eric von Hippel : The Sources of Innovation, Oxford University Press 1988
5. Eric von Hippel : Democratizing Innovation, MIT Press 2006
6. Everett M. Rogers : Diffusion of Innovations (Fifth Edition) Free Press 2003
7. James M. Utterback : Mastering the Dynamics of Innovation, Harvard Business School Press 1994
8. James Utterback, Bengt-Arne Vedin, Eduardo Alvarez, Sten Ekman, Susan Walsh Sanderson, Bruce Tether, Roberto Verganti : Design Inspired Innovation, World Scientific Publishing 2006
9. John D. Sterman : Business Dynamics Systems Thinking and Modeling for a Complex World, McGraw Hill 2000
10. Karl T. Ulrich, Steven D. Eppinger : Product Design and Development (Third Edition), McGraw-Hill/Irwin 2004
11. Marc H. Meyer : FASTPATH to Growth, Oxford University Press - 2007
12. Marc H. Meyer, Alvin P. Lehnerd : The Power of Product Platforms Building Value and Cost Leadership, Free Press 1997
13. Mark W. Maier, Eberhardt Rechtin : The Art of Systems Architecting, CRC Press 2002

6. Product reference

1. Intel Corporation: Intel® vPro™ Processor Technology
   http://www.intel.com/products/vpro/index.htm?iid=prod_desk+sc_vpro

Case Studies on Intel® vPro™ desktop and Intel® Centrino® Pro laptop technology

1. http://www.intel.com/business/vpro/pdfs/swcaac_prodeval_casestudy.pdf
2. http://www.intel.com/business/vpro/pdfs/mor_prodeval_casestudy.pdf
3. http://www.intel.com/business/vpro/pdfs/computacenter_tech_brief.pdf
4. http://www.intel.com/business/vpro/pdfs/wm_tech_brief.pdf
5. http://www.intel.com/business/vpro/pdfs/getronics_tech_brief.pdf
6. http://www.intel.com/business/vpro/pdfs/star_softcomm_solutions_brief.pdf
7. http://www.intel.com/business/vpro/pdfs/ness_tech_brief.pdf
8. http://www.intel.com/business/vpro/pdfs/tata_tech_brief.pdf
9. http://www.intel.com/business/vpro/pdfs/uitm_casestudy.pdf
10. http://www.intel.com/business/vpro/pdfs/kaz_prodeval_casestudy.pdf
11. http://www.intel.com/business/vpro/pdfs/vsnl_casestudy.pdf
12. http://www.intel.com/business/vpro/pdfs/tata_casestudy.pdf

13. http://www.intel.com/business/vpro/pdfs/siemens_solutions_brief.pdf
14. http://www.intel.com/business/casestudies/atos_solutions_brief.pdf
15. http://www.intel.com/business/casestudies/bmw.pdf
16. http://www.intel.com/business/casestudies/capgemini.pdf
17. http://www.intel.com/business/casestudies/asklepios_casestudy.pdf
18. http://www.intel.com/business/casestudies/bangkok_general_casestudy.pdf
19. http://www.intel.com/business/casestudies/evesham.pdf
20. http://www.intel.com/business/casestudies/ing.pdf
21. http://www.intel.com/business/casestudies/iss/johns_hopkins.pdf
22. http://www.intel.com/business/casestudies/providence_casestudy.pdf
23. http://www.intel.com/business/casestudies/foshan_prodeval_casestudy.pdf
24. http://www.intel.com/business/casestudies/sinopec.pdf
25. http://www.intel.com/business/casestudies/eds_solutions_brief.pdf
26. http://www.intel.com/business/casestudies/proof_of_concepts_dcs.pdf
27. http://www.intel.com/business/casestudies/bt.pdf
28. http://www.intel.com/business/casestudies/proof_of_concepts_fujifilm.pdf
29. http://www.intel.com/business/casestudies/GijimaAst.pdf
30. http://www.intel.com/business/casestudies/lanzhou_railway.pdf
31. http://www.intel.com/business/casestudies/unisys_tech_whitepaper.pdf
32. http://www.intel.com/business/casestudies/zenith_solutions_brief.pdf

7. List of figures and tables

10. Sample Use Case - Remote Asset Inventory in "UC Set 1.a" : Built-In manageability Intel® vPro™ Platform, Source: Intel Corporation
11. Sample Use Case - Remote Diagnosis and Repair in "UC Set 1.a" : Built-In manageability Intel® vPro™ Platform, Source: Intel Corporation
12. Usage models evolve during the user "Proof of Concept"
13. Sample macro architecture in end user setting (AS-IS) for "Proof of Concept"
14. Conceptual architecture to enable Usage Models Source: Intel Corporation
15. Sample macro architecture in end user setting (TO-BE) for "Proof of Concept"
16. Reference architecture evolves during the user "Proof of Concept"
17. User "Proof of Concept" reference stack is usually pre-production
18. Reference stack evolves during the user "Proof of Concept"
19. User "Proof of Concept" package is a managed set of three elements
20. User "Proof of Concept" package "POC Set 1.x" is a managed set of three elements
21. User "Proof of Concept" window for user engagement
22. Mapping adopter categories to user "Proof of Concepts"
23. Time bound approach to managing user "Proof of Concepts"
24. Link user "Proof of Concepts" of an innovation to its launch event
25. User needs translate to platform capability and features
26. User needs translate to platform capability in the planning cycle
27. User needs may translate to requirements change in current innovation
28. Types of collateral - quantitative versus qualitative collateral
29. Targeting type of diffusion enabling collateral to adopter categories
30. "Proof of Concepts" feedback impact current and future innovation

"End of Document"