Functional Paleontology: System Evolution as the User Sees It

Annie I. Antón
College of Engineering
North Carolina State University
1010 Main Campus Drive
Raleigh, NC 27695-7534
+1 919.515.5764
anton@burdell.csc.ncsu.edu

Colin Potts
College of Computing
Georgia Institute of Technology
801 Atlantic Drive
Atlanta, GA 30221-0280 USA
+1 404.894.5551
potts@cc.gatech.edu

ABSTRACT
It has long been accepted that requirements analysis should precede architectural design and implementation, but in software evolution and reverse engineering this concern with black-box analysis of function has necessarily been de-emphasized in favor of code-based analysis and designer-oriented interpretation. In this paper, we redress this balance by describing "functional paleontology", an approach to analyzing the evolution of user-visible features or services independent of architecture and design intent. We classify the benefits and burdens of interpersonal communication services into core and peripheral categories and investigate the telephony services available to domestic subscribers over a fifty-year period. We report that services were introduced in discrete bursts, each of which emphasized different benefits and burdens. We discuss the general patterns of functional evolution that this "fossil record" illustrates and conclude by discussing their implications for forward engineering of software products.

Keywords
Human-computer interaction (HCI); Measurement, metrics, and empirical methods; Reverse engineering; Requirements engineering; Software evolution.

1 PALEONTOLOGY OF SOFTWARE FUNCTION

Software development practice is moving toward a product-line perspective [DS99], in which software systems are configurations of desired functions, packaged as features or *services*. Services are generally collections of functions that are related to a delineated purpose or mode of use. Examples include “call forwarding” in telephone switching and “spell checking” in word processing.

Functional evolution is the phenomenon by which a system's services change over time. Experience suggests that the dominant form of functional evolution is the addition of new functions to a *baseline*, but existing services may become refined or specialized, and obsolete services may be *displaced*. However, we are unaware of any detailed studies of the functional evolution of software. Historians of engineering have investigated the evolution of structure and function in physical artifacts as diverse as the paperclip [Pet92], jet engine [Bas88] and buildings [Bra94]. Lehman and Belady [LB85] studied the evolution of OS 360 over many releases, addressing the evolution of its structure (size and number of modules, and the concomitant development effort), and they distinguish between various kinds of enhancement, but they do not report in detail on the kinds of services introduced over time.

Yet the availability of a public “fossil record,” a chronology of service evolution for a single system, or better still a family or community of interacting and competing products, has practical consequences: the ability to understand how a given system or product family is likely to evolve functionally in the future. The systems that Lehman calls E-type systems, co-evolve functionally in conjunction with their environments [Leh80, LB85]. Requirements volatility (customer-desired short-term functional change) has been identified as a principal obstacle to software development [Cur88]. Given how much more pervasive and important change processes are in software development than in other engineering fields, it would be valuable to know what kinds of functional changes occur as systems evolve. Only when these have been identified or predicted can architectural principles like information hiding, component-based design [AG97], and design patterns [Gam94] be applied to the implementation architecture to make it more adaptable.

We believe it is worth investigating whether systems evolve functionally in non-random and partly predictable ways. Such investigations require the development of a *conceptual base*, or theory of function and evolution, and empirical base or *fossil record*. The work we describe in this paper is an initial investigation into the definition of system function and value by analyzing the growth and evolution of services. Our aim is to provide a vocabulary for delineating, classifying and comparing the value of services as they evolve so that future system evolution can be more rationally anticipated and planned for.

Section 2 outlines our conceptual base. In the first part we develop a taxonomy of service benefits and burdens that derive from an analysis of knowledge and communication types. The profile or "shape" of benefits and burdens at a given point in a system's evolution is its functional *morphology* at that time. The second part of Section 2 addresses the evolution of functional morphology, introducing the concepts of epochs, expansions and cohorts and discusses to what extent these concepts are dependent on or autonomous of implementation and environmental driving forces. Section 3 describes a functional fossil record, the evolving functional morphology of domestic subscriber telephone services in a US city over a fifty-year period. We present a summary of the major changes and then dig deeper, applying the functional morphology to reveal the underlying pattern. Finally, in Section 4, we discuss to what extent the introductory conceptual base and limited fossil record can be used to predict patterns of functional evolution or future requirements volatility in communications-support software specifically, and software engineering more generally. We close by relating our analysis of services to abstractions such as use cases and function points, and other ongoing and planned investigations into functional morphology and evolution.

2 CONCEPTUAL BASE: FUNCTIONAL MORPHOLOGY AND EVOLUTION

Before discussing our functional "fossil record," we must introduce two sets of concepts: those used to describe the "fossils" (morphology), and those used to describe patterns of change (evolution).

Functional Morphology

By *functional morphology* we mean the overall profile or shape of benefits and burdens exhibited by a system at a point during its evolution. These are determined by the services that the system provides.

Services endow customers with the potential to achieve *goals* that they could not achieve or could achieve less adequately in their absence. For example, *Caller ID* lets telephone subscribers identify callers without answering. A service is appropriate if it supports a stakeholder's goals. Services that fail to support goals or that support minor goals at a cost that is too great to justify are deemed pointless, decorative, or "gold-plating".

Previous research in requirements engineering has developed a rich theory of goals and their refinement into system constraints and operations [Ant96, AP98, vLDM95], the obstacles that may block goals in the deployed system's environment and the secondary defensive and mitigation goals that arise to make the system more robust in the presence of such obstacles [Pot99].

A service provides a customer with the *potential* to achieve goals. How much benefit customers realize depends on many factors. The major impediments to benefiting from feature potential are the burdens that the service imposes on its users. For example, a service may only be feasible if its users possess certain capabilities. Thus, *Caller ID* is meaningless in the context of a subscriber telephone that is not *Caller ID*-enabled. At the very least, it should have a display capable of showing the caller's number. Burdens may involve extra setup activities, memorization or attention, or constraints on the customer's mobility or location.

Benefits

There are varying levels and types of potential benefits: core versus modulating, autonomous versus reactive, and amplified or qualified benefits.

Core and Modulating Services. The most important distinction is between *core s*ervices and second-order or *modulating* services. Zuboff [Zub88] introduced the term *infomating* (in contrast to automating) to refer to modulating services the benefit of which stems from the creation or transmission of knowledge about the system itself and its operations rather than the underlying application domain. Thus, knowing who is calling you without answering the telephone is a second-order benefit in which you as subscriber have more knowledge about and control over the state of the system. However, this service would be of little use if you were unable to speak to or hear the caller, since it is communication potential that provides the subscriber with the telephone's core benefit.

Information systems and communication-support services are designed primarily to create or transmit knowledge. Thus the *core knowledge benefits* of services are automated informing, notifying or the enabling of communication, whereas the "infomating" benefits mainly concern state awareness (such as the identity of the caller or the presence of recorded calls).

Autonomous and Reactive Benefits. The benefits just considered are all autonomous in that they are intended as benefits of the system without consideration of the burdens imposed by them or by other services. Some benefits, however, are *reactive*. They gain their value from the disutility or burdens of other services. If a service makes a subscriber too accessible, services that enhance subscriber privacy become required in reaction. When a service imposes action or cognitive burdens, new reactive services may be required, services that let the user organize or marshal information.

Amplified and Qualified Benefits. A service's benefit may be qualified or amplified by an enhancement of the service or the addition of a new service. Information may be created or transmitted faster, in richer media, more accurately or more broadly. Thus being able to make conference calls increases the communication potential of the subscriber because it is now possible to communicate

with more than one person simultaneously; the video telephone lets the subscriber see as well as hear the caller.

Burdens
Services are not necessarily valuable, because an intended beneficiary may incur burdens that outweigh the benefits. We divide burdens into two broad groups: the withdrawal or diminishment of an existing benefit, and burdens proper, which have their own categories as follows.

Mechanism. Sometimes a service requires user-visible mechanisms that are not necessary in the absence of the service. Any service generally requires new or augmented implementation mechanisms, but by "mechanism" burdens we mean only those visible to the user; for example, acquisition of special equipment.

Location. Some services require the beneficiary to be located in a specific place, usually because of the sensory modalities implied (e.g. you must be within earshot of an alarm).

Role responsibility. Some services require the beneficiary to incur responsibilities in a new role. Having to learn and remember codes and procedures are typical examples.

Setup. The set-up actions required to initiate a service are a special case of role-related responsibility. These may include specifying information that modulates other services (e.g. customizing reports or shortcuts) or defining key data (e.g. the addresses from which email messages may be discarded).

Functional Evolution
Functional evolution refers to changes in functional morphology over time. Two kinds of evolutionary pattern may occur when phenomena such as software services undergo change: gradual or saltationist evolution.

Evolutionary biology has been marked by recent controversies concerning the degree to which speciation is gradual (as predicted by Darwin) or whether it is saltationist, with comparatively sudden extinctions, expansions and radiations of taxa [Sob93]. Similarly, in the history of science, Kuhn [Kuh96] famously made the distinction between epochs of “normal science” in which the growth of knowledge was steady and conservative, and rare “paradigm shifts” in which the foundations of the science in question were rethought and reorganized. Nearer to home, Belady and Lehman [LB85] in their study of the evolution of large software systems, distinguished between the normal and gradual maintenance processes that occur between major software releases and the occasional redesigns that punctuate this gradual process.

Similarly, the functional evolution of a system or product line could be gradual or saltationist. Which it is is an empirical question that depends on the interpretation of a functional "fossil record" (see Section 3). In saltationist evolution, services are introduced in bursts or expansions that separate comparatively stable “epochs.” A set of new services introduced during an epoch (mainly at its beginning) form a *service cohort* and the set of services the cohort enhances (i.e. those existing at the end of the previous epoch) form the *service baseline*. Any services removed before the next epoch form a *displacement cohort.*

3 A FOSSIL RECORD: THE EVOLUTION OF TELEPHONY SERVICES

The previous section introduced a generic set of concepts for discussing functional evolution. In this section, we examine the evolution of a specific set of services over an unusually long period: the provision of subscriber-visible services in domestic telephony over the past fifty years. We now describe our “paleontological” method and findings.

Methodology
Only publicly available information about telephone services was used. Specifically, we tabulated the named services contained in the call guide (the introductory section) of the Atlanta telephone directories for the years 1950-1999. In many cases, these are familiar services and for some years the call guides included instructions for use. In the case of ambiguous services, interaction diagrams were created to specify the core service-use scenarios, and where possible the services were used and tested.

The number of services available in a given year is therefore a coarse measure of the functional "size" of the system at that time. Intuitively, some services seem more central or significant than others, but to assign different weights to features based on an *a priori* classification scheme rather than obtaining a raw count would beg the very questions we seek to answer. We take account of centrality by classifying some services as yielding core as opposed to modulating benefits.

Telephony-Specific Benefits and Burdens
The functional morphology described in Section 2 is generic and needs to be instantiated for a specific application domain or product line (see Table 1). There is some room for subjective interpretation here, but we have carefully adopted the generic functional morphology categories when positing specific categories for the case of domestic subscriber telephony. In this case, the core knowledge or information transfer made possible by the system is transparent to the system’s services. Thus the *core knowledge* benefits are simply the communication of any speech by the subscriber (the principal stakeholder with whose requirements we are concerned) and the receptivity of the stakeholder to communications initiated by others. Beyond simple calling and receiving of calls, telephony services provide additional communication and accessibility benefits too. These involve the broadening of media (e.g. video) as well as the timeliness of communication and accessibility (e.g. recording, forwarding). We characterize all these types of service as

providing communication or accessibility potential.

In addition, more advanced services modulate these benefits by making the subscriber or caller aware of information about the state of ongoing or previous calls. Thus the *modulating* benefits are simply activities which inform the subscriber via some cue, such as a call waiting beep. There are two main types of reactive benefit: those relating to *privacy* (which includes non-disclosure of personal information and freedom from interruption) and those relating to *organization* of information (e.g. lists of special numbers).

TABLE 1: Benefit Categories for Telephony Services

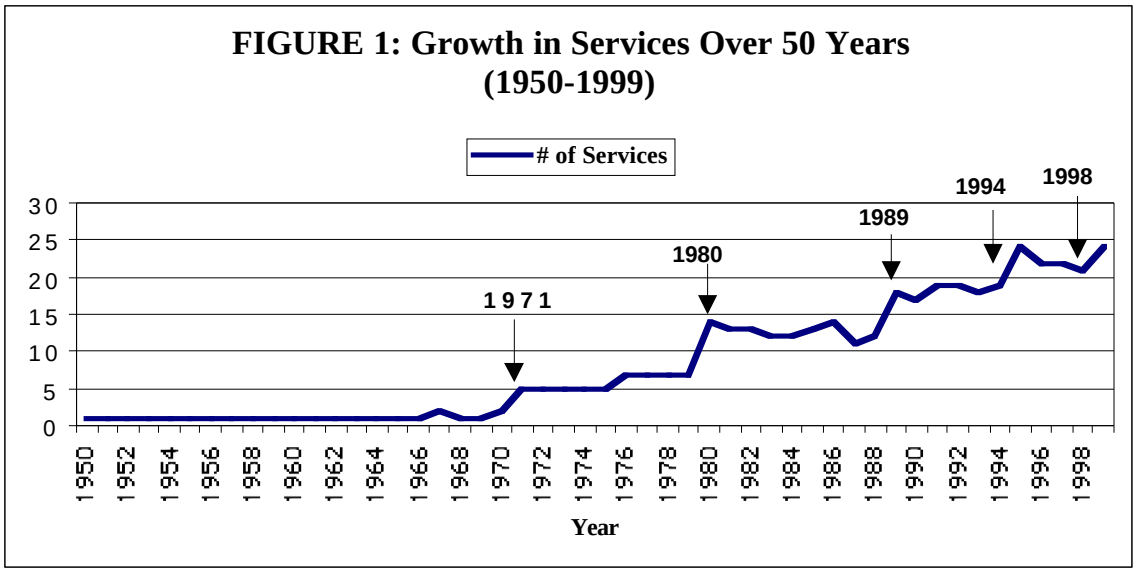
Benefit category / Generic type	Description	Example
Communication / Core knowledge	Communication with others beyond the potential afforded by the system's infrastructure.	Three Way Calling lets subscribers communicate simultaneously with two parties.
Accessibility / Core knowledge	Accessibility or availabilty to others' calls beyond the accessibility provided by the underlying service.	Call Forwarding Busy Line lets the subscriber be accessible even when the line is busy.
Awareness / Modulating	Enhanced knowledge about the context of a call or calls, including background information used to make decisions.	Call Waiting makes the subscriber aware that someone is attempting to call while another call is in progress.
Privacy / Reactive	Being freer from interruption, being less likely to disclose personal information or being more autonomous.	Anonymous Call Rejection offers both freedom from interruption and privacy since subscribers will not be disturbed when a caller with Line Blocking calls.
Organization	Organization of cognitive and external information resources.	Speed Calling maximizes cognitive resources by letting subscribers reach some numbers by dialing a single digit.

Analogously, we identify five types of telephony-specific service burdens (see Table 2). In the case of telephony, subscribers incur specific burdens, imposed by the service provider or the individual services themselves.

TABLE 2: Burden Categories for Telephony Services

Burden category / Generic type	Description	Example
Equipment / Mechanism	Being responsible for purchasing or providing specialized equipment to enable a service.	With the advent of Caller ID, special Caller ID boxes and telephones with built in Caller ID support became available.
Collocation / Location	Having to be in a specific place at a given time in the communication activity.	Call Forwarding requires that the recipient or a representative is at the forwarded location.
Action / Role responsibility	The need perform special actions during the normal span of the service.	Voicemail requires the subscriber to dial into the voicemail system and perform selection operations.
Cognitive / Role responsibility	Additional cognitive load, including short-term memory, discrimination of coded signals and decisions to act.	In the case of RingMaster, subscribers must distinguish among three distinctive ring patterns that signify the recipient. Call Waiting requires subscribers to decide whether they are available.
Setup / Role responsibility; setup	Special actions needed to initiate availability of the service but not normally required operationally.	These include subscription actions, programming of short cuts, etc.

Many services require subscriber's to purchase additional hardware or special apparatuses; for example, when the phone company stopped blocking subscriber identifiable information from callees, subscribers rushed to purchase new Caller ID boxes or Caller ID enabled phone units. We refer to such burdens as equipment burdens; other examples include TTY and video enabled phones. Collocation burden is also prominent in subscriber telephony since many services such as Caller ID, and more recently the Message Waiting Indicator, require subscribers to be physically located close enough to the phone unit to either read the LED display or glance at the light indicator. Telephony services that impose equipment and/or collocation burden force certain additional responsibilities upon their subscribers. Certain services require subscribers to perform a set of structured and infrequent actions to, for example, initialize and tailor a service's options (such as with Ring Master which requires subscribers to assign certain ring patterns to specific caller phone numbers) or more repetitive and routine actions (such as when subscribers interact with their voice mail service on a daily basis). Subscribers may also incur a cognitive or memory burden as is the case with services which require decision making (such as with Caller ID enabled Call Waiting which may require the subscriber to decide if the incoming call



more important than the current call already in progress) or discerning (such as required when a subscriber must be able to distinguish between specific ring.

Attributing Benefits and Burdens to Services

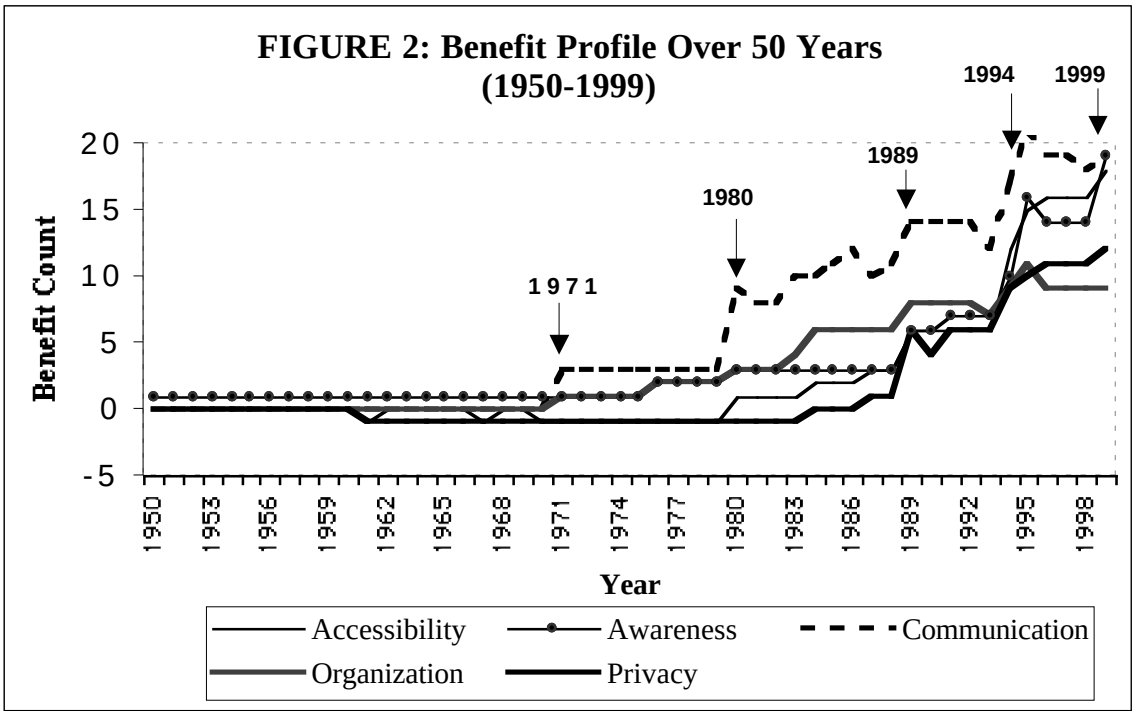
The preceding explanations and examples of benefit and burden types are general and intuitively plausible. Nevertheless, to attribute benefits and burdens more reliably to services, a more objective method is needed. The following method was used in the current study:

- Characterize potential benefits and burdens for a given service. These benefits and burdens must be additional to those that apply to the basic services of making and receiving calls. It is possible for a service to deliver more than one potential benefit of a given category or incur a several burdens of the same type. The resulting number of benefits or burdens of a particular type is the benefit (burden) *count* for that category for that service.
- Aggregate the benefit and burden counts for each category of benefit and burden over all the services available in a given year to obtain a set of benefit and burden counts.
- Chart the variation over time of the separate benefit and burden counts.

Results

Epochs and Cohorts.

Figure 1 charts the growth in services over time. Of the two feasible forms of evolution identified above, the



gradual and saltationist, the graph clearly shows the latter trend. Years of expansion are marked. Note that the number of services sometimes declines soon after an expansion.

Changes in Benefit/Burden Profile

More important than the number of services over time, however, are the *types* introduced. Table 3 provides more detail by identifying the service and displacement cohorts for each of the between-expansion epochs. Figure 2 combines these two presentations by showing the net benefit count over time for each benefit type. The same stepwise growth can be seen in Figure 2 as in Figure 1, but with the expansions affecting the benefit types differently.

TABLE 3: Service epochs and expansions

Epoch	Service Cohort	Displacement Cohort
1950-1970	Beep Tone	N/a
	Two Party Calling	
1971-1979	Collect	none
	Conference Calls	
	Person-to-Person	
1980-1988	Bill To 3rd Party	Beep Tone
	Call Fwding	Picture Phone
	Call Waiting	TDD
	Picture Phone	Two Party Line
	Speed Calling	
	TDD	
	Time/Charges Display	
	Three-Way Calling	
	Teletype	
1989-1994/5	Call Blocking	Restrict Calling Card
	Call Return	Sequence Calling
	Call Selector	TDD
	Call Tracing	Teletype
	Preferred Call Fwding	
	Repeat Dialing	
	Restrict Calling Card	
	Sequence Calling	
	TDD	
1994/95-1998	Block 900 (toll) Calls	Bill To 3rd Party
	Caller Id by Listing	Block 900 (toll) Calls
	Caller Id by Number	Caller Id by Number
	Call Fwding Always	CallFwdAll
	Call Fwd When Busy	Call Fwding Always
	Call Fwd on No Ans	Conference Calls
	Call Fwd Main Nmrs	Person-to-Person
	Remote Call Fwding	Time/Charges Display
	Ringmaster	
	Teletype	
1998-1999	Anon Call Rejection	none
	Beep Tone	
	Call Waiting Deluxe	
	Custom Code Restrict	
	Flex Call Fwding	
	Line Blocking	
	Msg Waiting Indicator	
	Voicemail	

For example, touch-tone service was introduced in 1980, and it spurred a major expansion during which the services Speed Calling, Three Way Calling, Call Forwarding and Call Waiting were introduced. These services greatly benefited communication potential (a core knowledge benefit) by providing more effective ways for subscribers to get in touch with others. The growth in communication benefits that year is attributed to the introduction of these services. A sharp increase in privacy benefits is evident later. This is partly due to the inception of a growing concern for privacy with the introduction of services such as Call Block, Call Selector and Call Tracing in 1989. These are reactive services that arose in response to the amplified communication and accessibility potential for the earlier expansion in core services.

Attributing Benefits and Burdens to Services

The preceding explanations and examples of benefit and burden types are general and intuitively plausible. To attribute benefits and burdens to services as objectively as possible, the following method was used:

- Characterize potential benefits and burdens for a given service. These benefits and burdens must be additional to those that apply to the basic services of making and receiving calls. It is possible for a service to deliver more than one potential benefit of a given category or incur a several burdens of the same type. The resulting number of benefits or burdens of a particular type is the benefit (burden) *count* for that category for that service.
- Obtain a set of benefit and burden counts by aggregating counts for each category of benefit and burden over all services available in a given year.
- Chart the variation over time of the separate benefit and burden counts.

Figure 3 charts the five kinds of service burden per year. Equipment burden is clearly not significant. Action burden is fairly linear and monotonically increasing, whereas cognitive burden is "bursty". There were two peaks in collocation burden, one in 1980 and the other in 1990. Recall that Call Forwarding was introduced in 1980 and this was the first time that customers were able to remain in communication and accessible while away from their home phone number. Of course, with this potential comes a collocation burden. For a subscriber to fully appreciate the potential of Call Forwarding, the subscriber must be physically present at the number to which their calls have been forwarded. Another service burden, setup, also experienced peaks.

4 DISCUSSION

Summary, Implications and Recommendations

Having a functional fossil record of a major application or infrastructure system is a valuable slice of knowledge in its own right.

We have claimed, however, that the case of software-controlled systems, such evolutionary data is likely to have practical, forward-engineering value arising out of the fluid nature of stakeholder requirements and the greater inherent malleability of software technology than that exhibited by physical systems. We now discuss the general phenomena to be found in the telephony functional fossil record and

their implications for software engineering development.

Punctuated Evolution

In Section 2, we distinguished between normal evolutionary growth and expansions. The evolution of the telephony services unambiguously show the primacy of punctuated evolution over gradual enhancement. While this may not be typical of software applications in general, it is remarkable that it occurred in the case of an infrastructure that undergoes continual "releases", with the services available to subscribers being documented annually. On *a priori* grounds, we might have expected the evolution of telephony services to be more gradual than that of applications that undergo major releases every year or two. Whatever the cause, and telephony professionals with whom we have discussed this are not short of technology-based and economic explanations for *specific* expansions, we draw attention to the more general lesson that *some* factors in the system's technical and cultural environment are likely to promote punctuated evolution. In the case of one system at one point in time these could include the de-regulation of the US telephone industry; in another case, they may include the growth of Internet-based commerce and changing concepts of privacy in society.

What this means in practice is that product development at any time is predominantly conducted in either "normal", incremental mode or "expansionist", saltationist mode. The distinction between major and minor releases or upgrades is enshrined in the informal numbering policies that many organizations use when releasing software, but largely as an administrative or configuration-control device. Given that external pressures are likely to drive the need for more and different kinds of benefits in discrete expansions, we suggest that the technical and management processes an organization uses during development should reflect this distinction explicitly.

Periodic Retrenchment

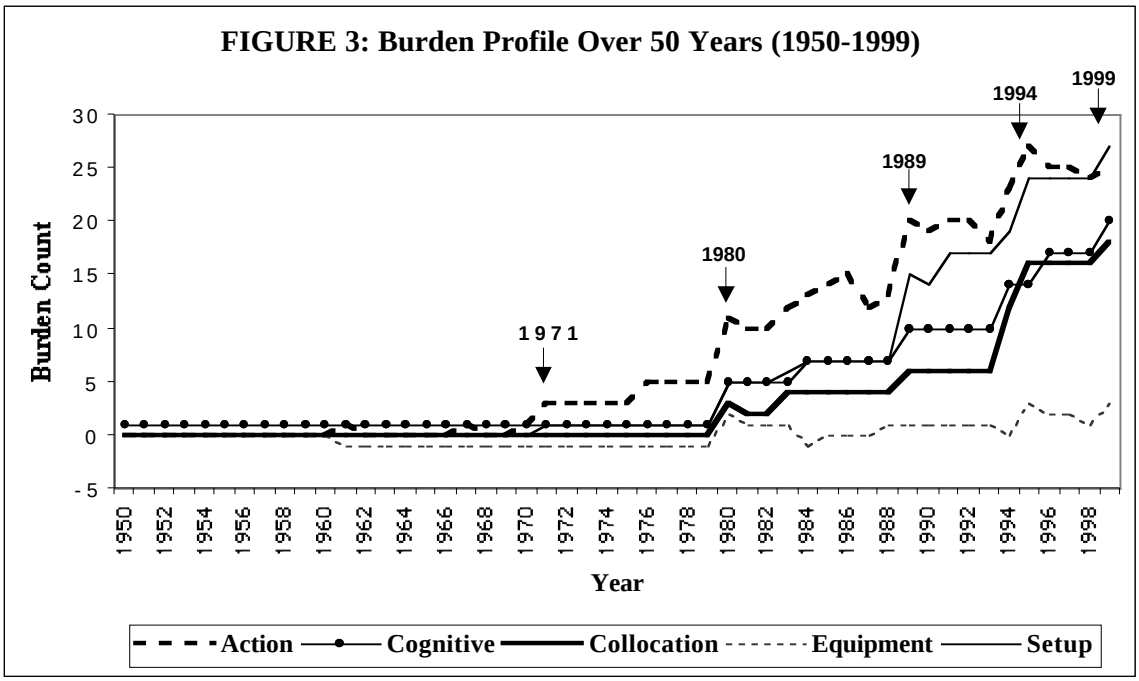
Expansions are usually followed by a small decline in the number of services and the benefit count. A diffusion explanation of this rebound effect is that technological opportunities make some services possible, which are subsequently consolidated when it turns out that they are not all necessary or are resisted culturally (e.g. being contested on regulatory grounds, in the case of telephony services). An evolutionary explanation is that some previous services are now seen to be redundant and the new ones displace them, as in the displacement of Caller ID by Caller ID Deluxe in which the caller's name is given rather than a telephone number.

Functional Decentralization

Not only was the evolution of telephony services punctuated, the expansions were different from each other in a systematic and generalizable fashion. The initial normal growth and expansions in services mainly emphasized those providing core benefits. While these always retained their role as core benefits (i.e. the telephone system is still all about communicating with others and being accessible to them), the later expansions included services that gain much of their imputed benefits from the inadequacy of, interactions among, or inventive abuse of earlier services.

The precise sequence of expansions is surely unique to a given system and is affected strongly by implementation concerns and the social and organizational climate in which the system is used. However, the evolution of telephony services suggests the following general trends:

FIGURE 3: Burden Profile Over 50 Years (1950-1999)



a) Benefits for the actor beneficiary precede benefits for others. Communicating with others dominated the first epochs in telephony; being accessible to others who wish to communicate with you, while significant from the start, only dominated later. Writers in HCI and social computing [Gru94] have emphasized that the features of an initially diffusing technology should benefit the people who actually perform tasks with it rather than those for whom they are acting as agents or with whom they collaborate indirectly. In a very different domain, the production of electronic presentations, one can see in products like Powerpoint, features intended for the author/presenter taking priority in early versions over features intended for the live or archival audience.

This suggests a somewhat Machiavellian requirements guideline that projects should be more careful in prioritizing and clarifying the requirements obtained from actor stakeholders, especially when system use is discretionary, than those obtained from other stakeholders.

b) Object-level benefits precede meta-level benefits. Where benefits refer to the creation and transfer of information (object-level benefit), such services take priority over services that add value to that information by creating or transferring information *about it* or about how the system is manipulating it (meta-level benefit). Thus awareness and organization benefits take off in the recent development of telephony services, and as telephony and electronic mail services continue to converge in the near future, we predict much greater emphasis on services yielding such benefits. Services providing awareness and organization benefits arise in reaction to the added complexity of the earlier core services. (One needs caller-id only if one must constantly decide whether to answer the phone.)

This suggests the relative unimportance, despite what customers or marketing specialists may indicate, of customization features in any product. Once the core services are largely in place, design options for meta-level services such as user customizations may be significantly limited by the legacy of core design decisions. Rather than arguing whether customization is good or bad in a software utopia, we simply observe that meta-level services are likely to be unimportant early in the development of most systems and constrained later. However, specific meta-level services are worth anticipating in advance where the core services greatly amplify access to a particular kind of knowledge or communication resource. These include lists of user-specific instances (e.g. friends' phone numbers, a personal dictionary), categories (e.g. types of message) or events (e.g. importance of calls).

Benefit/Burden Dialectic

Once a service has been designed and deployed it ceases to be an artificial creation and becomes part of the user's natural experience. Thus, irrespective of the designers' intentions, the service takes on a life of its own, its potential benefits may fail to be realized, it may be reinvented by users who adopt it in creative and unintended ways (e.g. the answering machine being used as a call-screening rather than recording device), and it may give rise to new modes of activity, including countermeasures to its perceived abuse. The literature on technology diffusion is littered with examples of unforeseen reinventions by users [Rog95], and Dahlberg & Mathiassen [DM93] make this "dialectic" between hopeful deployment ("thesis"), non-optimal use ("antithesis") and reinvention in the field ("synthesis") a major thrust of their design approach. More colorfully, Tenner [Ten96] talks of technology "biting back", when an acute benefit (e.g. agricultural pest control) gives rise to chronic burdens (e.g. rapid evolution of resisant pests) thus creating the need for further reactive innovations.

In the evolution of telephony services, we see this dialectic, or competition between types of benefit or burden, in the tensions between accessibility and privacy and in the general amplification of core benefits on the one hand and the concomitant need for awareness-enhancing and organizational services on the other. In both cases, the reactive services lagged the introduction of the services and the initial epoch of their use.

Benefits are always tarnished in some way, and we therefore recommend that new services always be analyzed in terms of benefits and burdens. Even though a trade-off analysis may clearly indicate the net benefit of a service now (and therefore that it should be given high priority), recognizing its downstream burdens gives the designer some confidence about what reactive services may be required subsequently and therefore should be planned for in the design of architectural structure. While the precise details will be specific, general patterns can be detected that do not require specialized domain or sociological knowledge. For example, amplified knowledge-creation and transmission benefits generally increase the likelihood that meta-level services will be needed later to marshal and make sense of the object-level information that the earlier services make available. Thus the need for improved filtering, user-interface and visualization features can be predicted in the expansion following the introduction of any powerful knowledge-creation service.

Finally, one stakeholder's benefit often becomes another's burden. We addressed this in passing earlier when emphasizing the core actors. We are currently investigating the benefit/burden profiles for different stakeholders in communications applications, which turn out to be subtle and difficult to anticipate. Most readers will be familiar, however, with the "arms races" between e-commerce services that provide convenience at the cost of disclosure and those that protect consumer privacy at the cost of denial of service.

Evaluation

Validity and Generality

The fossil record presented in Section 3 is based on a sample of service data for one region of the US. Wireless telephony, data communications, and business telephony services are among the services that we did not consider because of our data-gathering method. It is possible therefore that our data harbors systematic bias. (For example, perhaps business-oriented services would have revealed more gradual enhancement of core communication benefits.) But there seems no good reason to posit such bias other than the concern that the data is unavoidably incomplete.

Our concrete data applies to consumer telephony services, but there seems no good argument for supposing that the principal trends identified above are restricted that narrowly. Communication, accessibility, privacy, awareness, and organization benefits are general to all communication and coordination applications, so we are confident applying our general conclusions to group-based productivity and CSCW applications and infrastructures. Indeed, with minor modifications and a different instantiation of the general benefits identified in Section 2, the general conclusions from the previous section should apply to all dynamic information systems.

Our conceptual base and empirical conclusions seem therefore to apply to systems dominated by what Jackson [Jac95] calls the dynamic information systems problem frame and some variants of the control problem frame in which the phenomenon controlled by the system is the coordination of acts of human communication. Transformational applications such as compilers or pattern recognizers, physical control systems, such as manufacturing or avionics control systems, static information systems, such as search engines, and workpiece software products, such as word-processors or spreadsheets, deserve separate treatment. An example of a more object-oriented conceptual basis suited to workpiece applications is the investigation by Hsi and Potts [HP00] of the successive releases of a commercial office productivity suite.

Function, Structure and Environment

The treatment of functional evolution in this paper is unusual in our deliberate ignoring and downplaying of what is known about the underlying switching technology and business environmental factors during the period of evolution that we studied. A more complete treatment of telephony service evolution or the evolution of any application services, warrants the incorporation of knowledge from public policy, usability studies, cultural studies, and the details of the design architecture and enabling technologies in question. However, our perspective is that these issues are of secondary importance for an understanding of *why* functional evolution occurs. A system is first and foremost what it does, not what it is made from.

Related and Future Work

Benefits and Function-Based Measurement

The primacy of function over structure lies behind size measurements such as function points, which are weighted combinations of input- and output-attributes of a planned system [Mac94]. Function points are a more requirements-oriented basis for cost estimates than lines of code, but they do not directly address the value or usefulness of different services. Function points and benefit/burden count together provide a more detailed estimate of the functional "size" of a system or collection of services than either does alone.

Services Compared to Use Cases

Services are also closely related to *use cases* [Jac92], which are modes of system use that external agents initiate with a purposeful input. But services are not the same as use cases. Unlike use cases, a service's scenarios may exhibit temporally disjointed episodes. For example, in *Caller ID Deluxe*, a subscription episode initiated by the subscriber, occurs just once on service inception. Then, after an indeterminate interval, a caller initiates an identification episode, a type of episode that is repeated for every subsequent incoming call (Note that the identification need not be successful for the episode to occur.) A use-case analysis of *Caller ID Deluxe*, at least in Jacobson's original formulation, would differ markedly in making the subscription and calling episodes completely different use cases. Other services may involve more types of episode. For example, *Voicemail* involves subscription, calling and recording a message, retrieving messages, and even, arguably, replying to them.

Services and the Context of Activity

Services are not the only ways for a customer to achieve his or her goals. *Workarounds* are bundles of manual behaviors and the inventive uses of other automated services to achieve the same purpose, albeit often imperfectly and less conveniently. In the absence of *Caller ID*, a caller and recipient might agree on a code that identifies the caller, such as ringing twice in quick succession. Workarounds may be useful in the presence of service unavailability, excessive cost, or temporary breakdown, but they typically result in degraded outcomes and require additional responsibility, including, sometimes additional actions by agents other than the customer.

In Activity Theory [Nar96] tools are an integral part of the activity, shaping the way we conceive of and plan activities and learn skilled behaviors. What constitutes a "tool" in Activity Theory is more abstract than the everyday use of the term implies, since it includes any part of the world or culture, such as language, whose affordances lend it to being recruited to some purpose. Because services affect the behavior of agents and constrain their choices, services are tools in the Activity-Theoretic sense. As activities

change in the world (e.g. increased use by business of telemarketing) so the appropriateness of a service (e.g. Caller ID) changes and the adoption of other tools and behaviors will follow as a result of these changes. A service such as Caller ID may adapt the behaviors of a caller, as over time callers learn tactics for publicizing or hiding their identity.

Functional Morphology and Value Analysis

Numerous proposals have been made for assigning priorities to features or requirements and managing trade-offs involving multiple requirements and design criteria [Boe95, KR97]. Sullivan's [Sul96] application of real options theory to the valuation of product features is a more recent and economically more sophisticated contribution of the same kind. These approaches all have in common the provision of a normative basis for decision making and prioritization, a requirement that demands pairwise prioritization of features or, alternatively, the reduction of benefit to a common "currency" of comparison – either in terms of abstract utility quantities or by translation of stakeholder preferences into explicit monetary values.

In contrast, we argue for preserving the multidimensional and fundamentally incommensurable nature of different benefits and burdens if the aim is to understand the likely future consequences of introducing features and services. An additional advantage of doing so, which we are exploiting in continuing investigations of telephony services, is the ability to compare actual or proposed services with already available work-arounds or the creative use of existing services to approximate the same benefits.

ACKNOWLEDGMENTS

This research was supported in part through collaboration with BellSouth Telecommunications via the North Carolina State University Center for Advanced Communications and Computing. Staff of BellSouth Advertising and Publishing Company and the Atlanta History Center provided the authors with historical records of telephone service availability. Anonymous reviewers of a previous version of the paper made many helpful comments.

REFERENCES

[AG97] Allen, R. and D. Garlan. "A formal basis for architectural connection" *ACM Transactions on Software Engineering and Methodology*, Volume 6, No. 3, pp. 213 - 249, July 1997.

[Ant96] Antón, A.I. "Goal-Based Requirements Analysis," *International Conference on Requirements Engineering (ICRE `96)*, Colorado Springs, Colorado, USA, pp. 136-144, April 1996.

[AP98] Antón, A.I. & C. Potts. "The Use of Goals to Surface Requirements for Evolving Systems," *Int'l Conference on Software Engineering (ICSE '98)*, Kyoto, Japan, pp. 157-166, 19-25 April 1998.

[Bas88] Basala, G. *The Evolution of Technology*. New York, Cambridge University Press. 1988.

[Boe95] Boehm, B. P. Bose, E. Horowitz & M.J. Lee. "Software requirements negotiation and renegotiation aids", 17th Intl. Conf on Software Engineering, Seattle, WA, pp. 243-253, 24 - 28 April 1995.

[Bra94] Brand, S. *How Buildings Learn: What Happens after They're Built.* New York, Viking. 1994.

[Cur88] Curtis, B., H. Krasner, et al. "A field study of the software design process for large teams." *Communications of the ACM* 31(11): 1268-1287, 1988.

[DM93] Dahlbom, B. and L. Mathiassen. *Computers in Context : The Philosophy and Practice of Systems Design*. Blackwell, Oxford. 1993.

[DS99] DeBaud , Jean-Marc and Klaus Schmid. A systematic approach to derive the scope of software product lines; Proc. Int. Conf. Software Eng. Los Angeles, CA: 1999, pp. 34 - 43.

[Emm89] Emmerson, Andrew. "Strowger and the Invention of the Automatic Telephone Exchange: 1879-1912," 1989.

[Gam94] E. Gamma, R. Helm, R. Johnson, J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, Reading, MA 1994.

[Gru94] Grudin, J. *Computer-supported cooperative work: History and focus*. IEEE Computer: 19-26, 1994.

[HP00] Hsi, I. & C. Potts (in press) "Studying the Evolution and Enhancement of Software Features". *Intl. Conf. Software Maintenance*, San Jose, CA, 11-14 October 2000.

[Jac92] Jacobson, I. et. al. *Object-Oriented Software Engineering: A Use Case Driven Approach*, Addison-Wesley, 1992.

[Jac95] Michael Jackson. *Software Requirements and Specifications*. Addison-Wesley, 1995.

[KR97] Karlsson, J. & K. Ryan. "A Cost-Value Approach for Prioritizing Requirements". *IEEE Software*, pp. 67-74, 1997.

[Kuh96] Kuhn, T.S. *The Structure of Scientific Revolutions*, 3rd edition Univ of Chicago Press, 1996.

[LB85] Lehman, M.M. and L.A. Belady. *Program Evolution - Processes of Software Change*, Academic Press, London, 1985.

[Leh80] Lehman, M.M. "Programs, Life Cycles and the Laws of Software Evolution," *Proc. IEEE* Special Issue on Software Evolution, IEEE, Piscataway, NJ, *Vol.68, No. 9*, September 1980, pp. 1060-1076.

[Mac94] MacDonell, S.G., *Comparative Review of Functional Complexity Assessment Methods for Effort Estimation.* Br. Computer Soc./Inst. Electrical Eng. J., 1994. 9(3): p. 107-117.

[Nar96] Nardi, B.A., ed. *Context and Consciousness: Activity Theory and Human-Computer Interaction*.. 1996, MIT Press: Cambridge, Mass.

[Pet92] Petroski, H. *The Evolution of Useful Things*. New York, Vintage Books. 1992.

[Pot95] Potts, C. "Invented Requirements and Imagined Customers: Requirements Engineering for Off-the-Shelf Software." *Second IEEE Int'l Symposium on Requirements Engineering*. 1995. York, UK: IEEE Computer Society Press.

[Pot99] Potts, C. "A ScenIC: A Strategy for Inquiry-Driven Requirements Determination," IEEE 4th International Symposium on Requirements Engineering, Limerick, Ireland, 7-11 June 1999.

[Rog95] Rogers, E. M. Diffusion of Innovations. New York, Free Press, 1995.

[Sob93] Sober, E. The Nature of Selection: Evolutionary Theory in Philosophical Focus. Chicago, IL, Chicago University Press, 1993.

[Sul96] Sullivan, K.J., "Software Design: The Options Approach," 2nd Intl. Software Architecture Workshop, Joint Proceedings of the SIGSOFT '96 Workshops, San Francisco, CA, pp. 15--18, Oct. 1996.

[Ten96] Tenner, E. *Why Things Bite Back: Technology and the Revenge of Unintended Consequences*, Knopf, 1996.

[vLDM95] van Lamsweerde, A. and Darimont, R. and Massonet, P. "Goal-Directed Elaboration of Requirements for a Meeting Scheduler: Problems and Lessons Learnt," Proceedings 2nd International Symposium on Requirements Engineering (RE`95), York, UK, pp. 194-203, March 1995.

[Zub88] Zuboff, S. In the Age of the Smart Machine: The Future of Work and Power, Basic Books, 1988

.