



NORTHWESTERN UNIVERSITY

Electrical Engineering and Computer Science Department

Technical Report
NWU-EECS-08-14
January 25, 2008

Sustainable Development of the Internet

Leiwen Deng and Aleksandar Kuzmanovic

Abstract

In recent years there has been a great deal of research surrounding the architecture of the Internet. Despite this, researchers still do not share the same views on two basic questions: *What are the fundamental problems of current Internet? Is a new architecture needed or can we tweak the current Internet, solving problems through minor upgrades?* In this paper, we look at the Internet's architecture, design, and the underlying problems at a high level, considering axiomatic criteria of Internet development across multiple domains (technical, economic, and social). We observe that the fundamental problem of current Internet is its insufficient *sustainability*. We therefore propose a framework for a generalized, new Internet architecture that addresses both short and long term development issues.

Keywords: sustainable development, sustainability, identity, standards, incentives, auditing

Sustainable Development of the Internet

Leiwen Deng and Aleksandar Kuzmanovic

ABSTRACT

In recent years there has been a great deal of research surrounding the architecture of the Internet. Despite this, researchers still do not share the same views on two basic questions: *What are the fundamental problems of current Internet? Is a new architecture needed or can we tweak the current Internet, solving problems through minor upgrades?* In this paper, we look at the Internet’s architecture, design, and the underlying problems at a high level, considering axiomatic criteria of Internet development across multiple domains (technical, economic, and social). We observe that the fundamental problem of current Internet is its insufficient *sustainability*. We therefore propose a framework for a generalized, new Internet architecture that addresses both short and long term development issues.

1. INTRODUCTION

Sustainable Development is development that meets the needs of the present without compromising the ability of future generations to meet their own needs.

— Brundtland Report, “Our Common Future”.
United Nations’ World Commission on Environment and Development, 1987.

The Internet currently plays an important role in our daily lives and its impact continues to grow. But at the same time, it faces many problems. Numerous research studies on a new Internet architecture (*e.g.*, [13, 27, 30, 35, 36]) have addressed these problems from different perspectives. But till now, researchers still do not have a consistent view on both the new Internet architecture itself and the underlying problems that motivate it. This paper analyzes the Internet architecture and its underlying problems from a higher level perspective — axiomatic criteria of Internet development, which spans not only *technical* domain, but *economic* and *social* domains as well. From this new perspective, a clear-cut and consistent picture of the underlying problems shows up. Based on that, we propose a solution — a framework for a new, generalized Internet architecture that addresses both long and short term development goals.

Analyzing criteria of Internet development across the technical, economic and social domains, we find that the problems in the current Internet architecture stem from its lack of *sustainability* which impedes future development. This sustainability problem shows in two aspects: (*i*) *evolvability* issues and (*ii*) pressure on a set of *cultural norms* (*e.g.*, cooperation, trust, creativity, economic and social order) that Internet development relies on, which is similar to the central problem that traditional sustainable development tries to solve:

economy development puts pressure on the natural environment but also relies on it.

On one hand, evolution of the Internet is driven by new demands of services that run on it and is expected to meet the demands. The current Internet, however, does not have sufficient evolvability to keep up with new demands. For example: (*i*) There is no effective countermeasures against *distributed denial of service* (DDoS) attacks, which prevents many good service ideas from being applied in practice. (*ii*) There is a long anticipated demand for *quality of service* (QoS), but it is still far from being met due to architecture restraints.

On the other hand, Internet architecture should provide leverage points, by which we can direct Internet services to evolve in a sustainable way. To be specific, we should be able to make Internet services improve those cultural norms on which their development depends, or at least not deteriorate the norms if unable to improve. For example: (*i*) Global cooperation among service providers is an essential element for Internet development, but a design dogma today is that we are better off not assuming service providers’ willingness of cooperation. Can we find any leverage points to cultivate global cooperation? (*ii*) Copyright issue is another example of the norms. Copyright infringing objects (video, audio, software) exist in large quantities on the Internet yet there is no effective countermeasure. As a side effect, content pollution becomes justified in many cases, although it is annoying most of the time. (*iii*) The Internet’s pressure on economic and social order as well as human mental experience keeps increasing. Internet addiction becomes a non-trivial social issue¹. *Real money trade*² for virtual goods causes significant tension on economic and social order.

We argue that we can address both aspects by adding proper *controllability* and a *semantics aware property* to the Internet architecture. The controllability does not restrain freedoms; it instead fosters a higher extent of freedom resulting from advanced flexibility and functionality provided by the new architecture. A semantics aware property makes the entire architecture (from the highest to the lowest layer) evolvable based on service level semantics. To support the controllability and semantics aware property, we introduce four leverage points for our framework: *identity*, *standards*,

¹According to [4], more than one out of eight Americans exhibits signs of Internet addiction. The biggest culprit of Internet addiction is not online pornography, games and gambling as some people think, but ordinary services such as email, online chatting and shopping.

²Real money trade is the buying and selling of virtual items in online games for real world money.

incentives, and *auditing*. The four leverage points work in synergy, forming the base of our framework.

Our framework for a new Internet architecture learns and applies basic ideas and experiences from the traditional *sustainable development* [19]. We find most principles in that area can be directly applied to the Internet context if we treat the aforementioned *cultural norms* as the counterpart of the natural environment in the traditional sustainable development area. Moreover, we find that these principles can work better in the Internet context. Most surprisingly, applying them to the Internet context can create priceless products that lead to fundamental innovations not only for Internet development but also for scopes beyond it. A typical example of such products is the improvement of common human values and social trusts, which is the key to solve many problems that are currently unsolvable by technical means³.

Sustainable development challenges are huge as pointed out by principles of traditional sustainable development, but there is an *inherent advantage*: Progress in sustainable development creates new benefits and tools that help its further progress, *i.e.*, *sustainable development provides solutions to itself*. Therefore, what we need to do now is to find critical leverage points and make the “first snowball” (similar to what *GENI* [2] and *FIND* [1] projects are working towards). Once we get it, what we do next is to simply roll the snowball. Developing the “first snowball” of sustainable development is what we focus on in this paper.

The rest of this paper is organized as follows: In Section 2 we introduce basic theories in traditional sustainable development and explain how these insights can be applied in the Internet context. In Section 3 we describe our framework and design principles for the new Internet architecture. In Section 4 we discuss our solution to evolvability. In Section 5 we show implementation examples of our framework. In Section 6 we present related work. Finally, we conclude in Section 7.

2. BACKGROUND

2.1 Sustainable Development

In this section, we introduce the background of sustainable development. To avoid ambiguities, we use the abbreviation *SusEco* to denote the *traditional* sustainable development (that relates to economy and ecosystem), and use the term *SusInet* to denote sustainable development in the Internet context.

2.1.1 Basic Principles

It is increasingly recognized that we need to achieve *sustainable development* — development that not only

improves economic goals, but also advances *social*, and *environmental* well beings simultaneously. Still, until the 1980s, the overwhelming opinion was that there were inevitable and fundamental trade-offs among the three. Taking economy and environment for example, people believed that the more one promotes development and growth, the worse off the environment will be. Indeed, the belief was that there was little possibility to achieve significant win-win outcomes. However, progress in the *SusEco* domain in the last three decades has shown the feasibility to achieve a win-win-win goal by better balancing the short- and long-term needs and government leadership. The following are five well known *SusEco* principles:

1. “*Business is good for sustainable development and sustainable development is good for business.*” Business is a part of the sustainable development solution, while sustainable development is an effective long-term business growth strategy.
2. “*Good governance is needed to make business a part of the solution.*” Good governance provides solution to conflicts arising from the interaction between the short-term pressure induced by businesses’ financial goals and the emerging principles of sustainable development.
3. “*Access to markets for all supports sustainable development.*” Sustainable development is best achieved through open, transparent, and competitive global markets.
4. “*Cooperation beats confrontation.*” Sustainable development challenges are huge and require contributions from all parties — governments, businesses, civil societies, and international bodies. Confrontation puts the solutions at risk. Cooperation and creative partnerships foster sustainable development.
5. “*Thinking locally, acting globally.*” While there is much we can do locally, action is also needed at the global level. There is an inevitable need for nations to collaborate to solve common problems.

We apply all the five *SusEco* principles to *SusInet*. Principle 1 points out *basic incentives and necessity* for ISPs and IT corporations to conduct *SusInet* practices. Principle 2 validates our approach to introduce *government involvement* (Section 3.6) into our solution. Principle 3 is the reason why we emphasize the *market-based approach* (Section 3.6.2). Principle 4 corresponds to a general solution to *SusInet*, *i.e.*, *Pyramid process* (Section 3.7). Principle 5 is straightforward due to Internet’s global nature. However, in this paper we focus more on *local acts* of nations.

2.1.2 The “Tragedy of the Commons” — Call for Control

The *Tragedy of the Commons* is a problem that has been known for centuries and still has no solution. It is a typical example that shows how *uncontrolled freedom* can lead to a disaster. The Tragedy happens when it is impossible, or at least very costly, to deny access to certain *common resource* (*e.g.*, marine fish). In a situation where many have access to the same limited resource, there is an incentive for each consumer to acquire as much of that resource as possible (*e.g.*, overfish), before

³Technical means are solutions that require “a change only in the techniques of the natural sciences, demanding little or nothing in the way of change in human values or ideas of morality” [21]. An example of a problem unsolvable by technical means is the *Tragedy of the Commons*, which has been known for centuries and still has no solution. It shows how uncontrolled freedom can cause a tragedy.

others do. This leads to overuse of natural resources in the SusEco context.

In the SusInet context, we observe essentially the same mechanism though the phenomenon is slightly different. Rather than suffering from the overuse of limited resources due to selfishness, the current Internet is more susceptible to malicious attacks (*e.g.*, DDoS) that deliberately deplete common resources (bandwidth, CPU power, memory, *etc*) despite the fact that many resources are over-provisioned. It shows essentially the same causality: since it is impossible to deny access to a common resource, disaster strikes.

Another analog of the Tragedy is the widespread copyright infringement, which shows similar causality: since it is impossible to deny access to infringing objects, copyright is overridden. In our SusInet solution, we address the root cause of the Tragedy. We add comprehensive controllability to the Internet architecture which can deny undue use of Internet resources.

2.1.3 The Role of Government

One central theory in SusEco is government involvement [19]. The theory points out the significance of good governance and the correct role of government in sustainable development. The critical role of the government in the SusEco context is to *address market and information failures*. There are two major cases where market fails: (i) The market price fails to include the hidden costs of *externalities*⁴, which are difficult to quantify due to their very nature. (ii) The market fails to distribute resources effectively in the *Tragedy of the Commons* scenario.

The theory also points out that government should emphasize on the *market-based approach* rather than *directly intervening* in market's basic functionality.

An Example of Direct Intervention: *To address the tragedy of commons, some governments practiced transferring certain property rights (e.g., that of forests, pasture land, in-shore fisheries) to a single independent body responsible for managing the public commons. However, extensive research and experience since 1968 shows that such transfers were sometimes disastrous for the resources they intended to protect. For the rest, many failed to do better than the market. Such failures result from: (i) High cost to manage, particularly when large numbers of individuals are involved. (ii) Not enough trained personnel on the ground to monitor resources. (iii) Corruption. (iv) Political climate sometimes roll back the role of government.*

The market-based approach, in which government provides *incentives and market signals*, minimizes risks. Still, practising such an approach is not easy. Actually, implementing incentives that motivate people to do the right thing and to make the right choices is one of the central challenges of economics. Despite challenges, the power of the market-based approach is huge. By sending right signals to the market, it can turn *vicious cycles* into *virtuous cycles* without additional expense [19].

⁴For example, the externality of smoke pouring from factories and fireplaces had many hidden costs for the economy, such as extra laundry cleaning or repairing corroded buildings.

2.2 Distinct Features of the Internet

The Internet has three distinct features: (i) *Global nature*. Internet provides global connection all over the world. (ii) *Fast service evolution*. Services on the Internet evolve much faster than the traditional business. (iii) *Fast information dissemination speed*. The Internet provides the fastest way to disseminate information — in particular, information that can attract interest of the majority. These distinct features create a specific environment which differs a lot from that of the traditional business. Such differences provide both extra challenges and opportunities to conduct sustainable development in context of the Internet. We argue that the opportunities surpass the challenges.

Challenges. One of the largest challenges for the Internet is the difficulty to *enforce the law*, which complicates the enforcement of the social norms and order that the Internet needs. This is due to both the *complex nature* and the *fast evolution pace* of Internet services, which make cyber-law enactment unable to keep up with. Even if the law enactment were responsive, its efficacy could be limited if the majority, or at least a large percent of users, violate the law. Meanwhile, it could take a long time to popularize new laws such that the majority obey them. Another major challenge is that Internet based businesses face more severe *short-term pressure* than traditional businesses, and such pressure could *last long*. This is due to both the more fierce competition resulting from the relaxed geographic restriction and the more likelihood that “early birds” can form a long-term global market dominance.

Opportunities. However, the Internet (in particular its fast information dissemination speed and global nature) provides good opportunities to address the above challenges. In addition, the Internet can significantly facilitate the implementation of the basic sustainable development principles. For example: (i) We can *accelerate the popularization* of laws and standards via passive and active learning mechanisms. (ii) We can enforce laws and standards globally (within each nation) by adding new features (Section 3.5) to the Internet. (iii) We can build an effective *unified incentive system* (Section 3.6.3) on the globally connected Internet. (iv) We can facilitate a general solution to the sustainable development (both SusEco and SusInet contexts), *i.e.*, *Pyramid process* (Section 3.7), by building an Internet platform to foster this process.

3. FRAMEWORK AND PRINCIPLES

In this section, we describe our framework and design principles of a new Internet architecture towards the sustainable development goal.

3.1 Framework Overview

At a high level, we address sustainability by adding proper *controllability* and a *semantics aware property* to the Internet architecture. We define two types of controllability: (i) *high level controllability*, which is the controllability of sustainable development, *i.e.*, the abil-

ity to direct Internet development to follow sustainable approach. (ii) *low level controllability*, which is the controllability of Internet access. The current Internet is a *default-on* network, *i.e.*, access is permitted by default. We address the low level controllability by adding a global *default-off* network [11,31,36] (*i.e.*, access is denied by default, Section 3.2) to the Internet. The semantics aware property means that all evolvable parts (that evolve as a result of new service level semantics) of the network architecture are able to capture related service level semantics. We achieve this property by introducing a new network layer model (Section 3.3).

The high level controllability is the core of our framework. We address it by introducing four critical leverage points: *identity* (Section 3.4), *standards*, *incentives* and *auditing* (Section 3.5). The four leverage points work in synergy, forming the base of our framework. *Identity* is a key to implementing our default-off network and is also the premise to build a *unified incentive system* (Section 3.6.3), in which we bind extensive information of a user to her identity. *Standards* exploit the semantics aware property and are the key to directing Internet evolution. However, both enforcement and enactment of new standards are non-trivial tasks. We exploit *incentive* mechanisms and *auditing* to help enforce standards. And we use *identity* as an ultimate weapon of enforcement. For standards enactment, we take advantage of the *Pyramid process* (Section 3.7), which is also a general solution to SusInet.

A central theory of SusEco is government involvement [19]. The theory points out the significance of good governance for *incentives* and gives comprehensive guidelines for the proper role of government in sustainable development based on experience of SusEco practice in the last three decades. We apply this theory to SusInet and revise the role of government according to Internet’s unique features (Section 3.6).

3.2 Abstract Structure of the New Internet

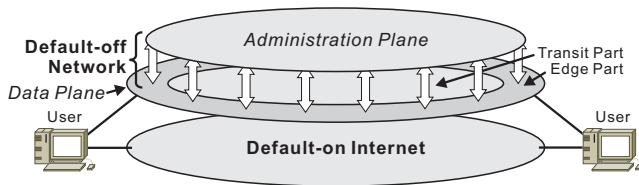


Figure 1: Abstract Structure of the New Internet

Figure 1 depicts the structure of the new, generalized Internet architecture that we propose. We are adding a default-off network to complement today’s default-on Internet. The default-off and default-on networks are logically separate networks which only merge at edges via end users. The default-off network consists of two separate planes: the *data plane* and the *administration plane*. The data plane is where user data is actually transferred; the administration plane provides centralized administration for the data plane.

The data plane is divided into the edge part and the

transit part. Routers are therefore classified as edge routers and transit routers. Transit routers are responsible for routing network traffic from edge to edge. Each edge router consists of two parts: (i) *routing part*, which is equivalent to an outermost transit router; (ii) *access part*, which works as an inverse firewall that performs access control by contacting the administration plane. Although the transit routers can contact the administration plane as well, our default-off solution does not rely on such communication. This allows the maximum freedom for designing the transit part (any implementation would be acceptable, *e.g.*, any routing mechanisms or multiple routing architectures in parallel).

3.2.1 Administration Plane

The administration plane is controlled by a single trusted third party, the government, as we will discuss in Section 3.6. The administration plane provides three basic services: (i) *Identity authentication*, which is the fundamental service that brings about a default-off network and is subject to be used massively. Therefore, its scalability is a major concern. In Section 5.1.2, we show an example implementation of identity authentication that takes advantage of large-scale replication to provide high scalability. (ii) *Unified incentive system*, which supports aforementioned critical leverage points. It gives individual users *incentive* to comply with *standards*. It is also an essential support for public *auditing*. (iii) *Distributed announcement database*, which is designed to announce authoritative information to public. It is a very useful common service.

3.2.2 Capability and Keeping State

Identity Based Capability. In our framework, we exploit *capabilities* [8,26,31,36] to implement the default-off network. To access the network, a user must first acquire a capability. Our capability solution (Section 3.4) is implemented based on a user’s permanent identity. This provides two distinct advantages: (i) It is *routing independent*, which we introduce in Section 3.4.1. (ii) *Keeping state* at network devices now becomes readily feasible and the network design therefore benefits from extraordinary flexibility. Currently, denial of service (DoS) attacks pose a significant threat to network devices that keep state. However, in our framework, we can apply identity based rate-limiting to effectively counter DoS attacks and exploit certain proof-of-work⁵ solution (*e.g.*, [26]) to solve the bootstrap issue⁶. In addition, with fear of consequences enhanced by traceable user identity, DoS attacks are fundamentally deterred.

Benefits of Keeping State. One of the essential benefits of keeping state is the design flexibility of control information. For example: (i) We can easily support *variable length or long control information*, which is

⁵Proof-of-work is asking service requesters to perform certain amount of computational work before providing service.

⁶An axiom [8] for capability-based solutions is that they rely on non-capability-based solutions for their bootstrap, *i.e.*, to establish the capability. Recent research [26,31] has given good solutions for this bootstrap issue.

problematic⁷ in the current Internet. (ii) We can safely use *out-of-band* channels (which are separate from but associated with the data channels) to transfer control information. (iii) We can significantly improve processing efficiency of control information by *amortizing the overhead* across entire service sessions.

3.3 New Network Layer

To support sustainability, we introduce a new network model for our default-off network. As shown in Figure 2, our new model is adapted from the TCP/IP model by revising the network layer. The new network layer has three distinct features: (i) *Semantics aware*. It is aware of service level semantics (*i.e.*, *abstract* application layer semantics) and handles traffic based on it. (ii) *Bidirectional extensibility*. It provides good extensibility in both upward and downward directions. The TCP/IP model instead only supports good extensibility in upward direction. A typical example of the downward extensibility in our model is that it can easily upgrade the network routing architecture and its underlying data link infrastructure; it can even support multiple routing architectures working in parallel. (iii) *Dual Layers*. It integrates both the transport layer and the network layer of the TCP/IP model. This necessarily results from the semantics aware property.

3.3.1 The Four Sub-layers

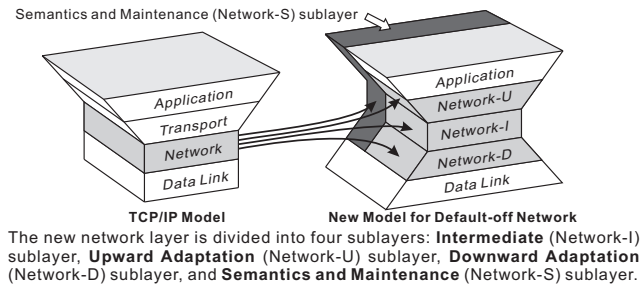


Figure 2: New Network Layer

We divide the new network layer into four sublayers: *network-I*, *network-U*, *network-D*, and *network-S* sublayers, as shown in figure 2. Such division decouples the variant part (subject to evolve) of the network layer from its invariant part (evolves very slowly, nearly immutable). The variant part includes the network-U, network-D and network-S sublayers while the invariant part is the network-I sublayer. Such decoupling allows for good evolvability as we will discuss in Section 4.

The *intermediate* (network-I) sublayer defines *abstract* and *immutable* primitives (network-I primitives) for network layer functionalities that are applicable to most application level services and semantics.

The *upward adaptation* (network-U) sublayer adapts network-I primitives to service specific protocols. In

addition, it defines new functionalities to meet the demands of evolving application layer services.

The *downward adaptation* (network-D) sublayer adapts network-I primitives to protocols specific to different network infrastructures. Our framework allows multiple network infrastructures to work in parallel (*e.g.*, one for timely services, one for bandwidth or reliability guaranteed services, one for best effort services, *etc.*). Based on service level semantics, the network layer redirects user traffic to the proper infrastructure.

The *semantics and maintenance* (network-S) sublayer converts application specific semantics into abstract canonical forms, based on which the network layer can process semantics efficiently. Meanwhile, it defines maintenance primitives and exposes a *common and compact* interface⁸ for maintenance functionalities (*management, fault diagnosis, measurement, etc.*). Any application and all layers (except data link layer) can convert their specific maintenance functionalities to network-S primitives, which are globally understandable.

3.3.2 Network Layer Metadata

The network layer *metadata* is the control information associated with a service session. It plays a similar role as the IP and TCP headers in the TCP/IP model, but supports more comprehensive functionalities.

One distinct and fundamental functionality of the metadata is to carry semantics and maintenance information. Semantics information is a central support for *standards*⁹ in our framework. Maintenance information is information exchanged by network devices for purposes such as management, fault diagnosis, and measurement. The network-S sublayer converts diversified semantics and maintenance information into canonical forms. Some of such canonical form information is then encoded into the metadata, allowing network devices to process associated sessions based on it. We exemplify the use of metadata in Section 5.2.

3.4 Identity

Identity is a key to implementing our default-off network. We use *identity based capability* to control network access. We introduce two forms of identity: *user identity* and *data identity*. Capabilities are issued through identity authentication which mandatorily inspects user identity and optionally checks data identity. User identity is also the premise for us to build user profiles such that we can apply effective incentive mechanisms.

3.4.1 User Identity

⁸As pointed out in [10, 12, 22]: (i) the difficulty of network management today is partially attributed to too many details exposed by heterogeneous infrastructures and diversified protocols at different layers; (ii) fault diagnosis can benefit a lot from correlating diversified elements at different layers. The network-S sublayer provides a common and compact interface to significantly reduce complexities resulting from the ever-evolving data plane.

⁹As we will explain in Section 3.5.1, standards define how to encode related semantics in the metadata and how end hosts and routers should process the metadata.

⁷When keeping state is not applicable, control information has to be piggybacked on every data packet. As a result, there are problems not only in processing efficiency but also in the fundamental feasibility of such control information.

Here we describe the design criteria for user identity. We show an implementation example that follows these criteria in Section 5.1.1.

Unified Identity. A capability is issued based on a user's permanent identity. The permanent identity is the same for all services on the default-off network, allowing for the maximum effectiveness of the capability. The unified identity can serve as an effective threat against misbehavior due to fear of consequences (access to all services could be affected). The unified identity also makes possible a unified incentive system (Section 3.6.3) that encourages users to behave well.

Use Single Trusted Third Party. Our unified identity solution uses a single trusted third party, an *identity authority*, which performs identity authentication and manages identity related information. In our framework, the identity authority is a government body.

Routing Independent Capability. Our capability solution allows the maximum freedom in the design of the routing architecture because it is not restricted to a specific routing mechanism. The routing part, *i.e.*, transit routers, work independent of capabilities.

Retain Privacy. Our identity solution provides traceability of a user's real identity. However, to be practical, it must be able to retain a user's privacy as well. To be specific, a user's real identity must be *untraceable* by unprivileged users. Here "untraceable" means: (i) *unresolvable*, *i.e.*, a user's real identity is not resolvable; (ii) *undistinguishable*, *i.e.*, unable to distinguish whether two observed identities belong to a same user.

Address Identity Theft. Identity theft is common in the real world. We must carefully address this issue: (i) A stolen identity must be able to be reclaimed by its owner. (ii) We need to reduce the impact of identity theft by both minimizing chances of it happening and restricting the negative impact on its owner if it happens. (iii) A proper mechanism must exist to let people quickly detect identity theft and respond to it. The quick response is desirable because it helps to suppress many kinds of misbehavior on the Internet that rely on identity masquerading.

3.4.2 Data Identity

In addition to user identity, we propose the idea of *data identity*, in which each data object is assigned a unique identity. This allows services to perform data-object-based authentication in addition to user-based authentication. We exemplify the usefulness of this feature in Section 5.2. In this section, we focus on the format of the data identity.

Three Basic Components of Data Identity. We can assign each newly created data object a new data identity. This data identity encodes the identity of the user who creates the object, the length of the object and the data fingerprint (digital fingerprint of the data object). The length and the data fingerprint encode unique properties of a data object such that we can verify the consistency between the data identity and the object.

Signature. A data identity also includes a digital sig-

nature by the identity authority which helps to verify the integrity of the three basic components. Before creating the signature, the identity authority verifies (via identity authentication) that the user who is requesting the signature is the same person as the user encoded in the data identity.

3.5 Standards, Incentives and Auditing

In order to guide Internet users (both individuals and businesses) to behave well, we need methods that can evaluate and respond to users' behavior in a qualitative and quantitative way, as cyber-law does. However, as pointed out in Section 2.2, cyber-law can not keep up with the fast evolution pace of Internet services. Therefore, we need a more responsive solution. In addition, at a high level, we need a mechanism to guide Internet evolution in the right direction.

3.5.1 Standards

We introduce *standards* to address both above issues. At the low level, standards work as "responsive cyber-laws". At the high level, standards bring about new protocols that reach global agreements among service providers. In this way, standards facilitate controlled Internet evolution.

Compared to cyber-law, standards are much more flexible due to the following two properties: (i) *More relaxed.* Standards enforce users to behave well in a more relaxed way than cyber-law does. Rather than addressing how to punish violations, standards emphasize on how to encourage and reward good behavior. Although standards also define penalties for violations, they are usually simply the deduction of benefits. (ii) *More tentative.* Since standards are more relaxed, they can be enacted more tentatively. Compared to cyber-law, we can introduce more tentative clauses to standards. If undesired side effects show up, the standard can be modified, suspended, or even canceled.

The tentative nature of standards allows them to be far more responsive than cyber-laws. And once a standard reaches maturity, it becomes a law. This benefits cyber-law enforcement by saving a lot of time for popularizing a new law because it has already become popular when we test the corresponding standard.

Many standards can be directly implemented by exploiting network layer metadata. In such cases, standards themselves are protocols. They define how we should encode semantics in the metadata and how end hosts and routers should process the metadata. However, some standards may require additional protocol upgrades at the network-U, network-D or network-S sublayers. This requires globally consistent upgrading operated by different service providers. In both cases, standards must be comprehensively enacted and be agreed upon globally by service providers and related entities. Nevertheless, it *must not* take too much time to enact since responsiveness is crucial. We can use the Pyramid process (Section 3.7), a tool adapted from SusEco to SusInet, to well address both the comprehensiveness and the responsiveness of standards enactment.

3.5.2 Incentives

Incentives are a central topic in sustainable development. Neither businesses nor individual users spontaneously follow the sustainable practices: In SusEco, such practices address the pressure on natural environment; in SusInet, they address the pressure on a set of culture norms that Internet development relies on.

In [19], the authors point out the significance of government involvement in creating incentives. They also give comprehensive guidelines for the proper role of government, which summarize the latest most significant developments within SusEco practices. We apply these guidelines to Internet development and find them well suited for the SusInet context.

To address incentives for businesses, we directly apply a guideline from SusEco (by treating the set of culture norms as the counterpart of the natural environment). The guideline is: Government should focus on the *market-based approach* to motivate businesses to follow sustainable practices. We discuss the market-based approach in Section 3.6.2.

To address incentives for individual users, we can exploit the relationship they have with businesses. A business that follows sustainable practices is very likely to guide its customers to follow such practices as well. Therefore, the problem can transform to the incentives for businesses. Moreover, as a second solution, we adapt the market-based approach for businesses to create a version that can be applied to individual users. This version of market-based approach is the unified incentive system (Section 3.6.3).

3.5.3 Auditing

To enforce standards, we must be able to verify whether or not a user's behavior comply with the standards. *Auditing* [9,20,28] is one of the solutions. Here we discuss the design principles of auditing.

Impossibility of Auditing. The ideal auditing mode is the one in which we can provide irrefutable evidence of a user's compliance with or violation of the standards. However, ideal auditing is extremely hard or even impossible to implement due to various difficulties: (i) Audit without infringing on users' privacy and data confidentiality. (ii) Audit without introducing too much overhead and cost. (iii) Audit without losing generality (e.g., make an audit method compatible with different routing mechanisms and path properties — regardless of whether it is single or multi path, reliable or lossy).

Three Strength Levels of Auditing. Due to the "impossibility of auditing", we relax the concept of auditing and define three different auditing types ranked by their strength levels:

Level-1 Auditing. It can not only detect but also provide irrefutable evidence for compliance or violations. This is the ideal auditing, *i.e.*, the strongest level.

Level-2 Auditing. It can detect both compliance and violations, but does not ensure irrefutable evidence. This type results from the fact that it can be extremely hard and costly to provide irrefutable evidence for *violations*

in some cases even though we can detect them. This is because users could try everything possible to disrupt the auditing in order to get away.

Level-3 Auditing. It can detect (and verify) compliance but not violations. Although this is the weakest level among the three, we argue that it is the most useful auditing type for SusInet. Because users have incentives to cooperate with such auditing by providing proof of their compliance, level-3 auditing is much easier to implement and can be used widely.

Cyber-Auditing and Classical Auditing. In addition to the three strength levels, we define two classes of auditing based on the amount of human involvement they require.

Cyber-Auditing. It is fully automated auditing that exploits computer technology and Internet support. It requires minimal human involvement. In our framework, the network layer metadata can facilitate the design of cyber-auditing methods because the metadata encodes service level semantics related to standards. However, the design and implementation of cyber-auditing could still be difficult, and in practice, we should rely heavily on the second class — classical auditing.

Classical Auditing. It relies heavily on human involvement and intelligence far beyond that of computers. It is modeled off of the financial auditing and the investigation in law. It is not restricted to computer or Internet based approaches. For example, it can collect clues and evidence via traditional approaches such as mail, phone, media, and personal inquiries, *etc.* However, the Internet can aid in classical auditing by facilitating information search. In addition, our new Internet architecture can significantly improve quality of information on the Internet since our controllability can substantially improve user responsibility.

Design Guidelines of Auditing. Here we summarize the nine design guidelines of auditing.

Incentives Rather than Censorship. Level-3 auditing, *i.e.*, to audit compliance only, is the most practical type of auditing. For level-3 auditing to be effective, the following rule should be used: Provide users incentives to comply rather than applying censorship.

Audit "To Public" Services. "To public" services which establish service sessions to public are relatively easy to audit (even level-1 and level-2 auditing is possible). We can effectively audit "to public" services at *endpoints* of the Internet.

Audit Individuals. Individual users are usually the hardest to audit for reasons such as possible collusion between source and destination endpoints, difficulty to provide irrefutable evidence of violations, *etc.* Therefore, to audit them we should emphasize on level-3 auditing which recognizes and rewards their compliance.

Audit Large Groups. Auditing large groups (e.g., big businesses and organizations) may not need an innovative approach. Classical auditing usually works well.

Exploit Commercial Relationships. We can take advantage of commercial relationships in auditing. In

places where such relationships (*e.g.*, customer-provider relationship) already ensure certain behavioral integrity, auditing mechanisms can be significantly simplified.

Public Patrol. We can have professional companies to play the role of *Internet patrols* (to audit Internet users' behavior¹⁰). However, since standards are tentative, such a job (Internet patrol) may become "volatile", and such a solution may not meet requirements of SusInet. We introduce the idea of *public patrol* (Section 5.2.4) to address this. It exploits the market-based approach to effectively allocate human resources for Internet patrol.

Audit Transit Routers. Methods to audit transit routers should focus more on measurement and fault diagnosis, which help evaluate routers' performance in relation to service level agreements among ISPs.

Audit the Government. The public should be able to audit government such that we can quickly detect and respond to any government (intentional or unintentional) wrongdoings and prevent abuse of power.

"No Major Impact" Principle. Level-3 auditing is insufficient in cases where the entire system can be influenced by a small minority of users in violation of standards. For such cases, level-2 or even level-1 auditing is imperative.

3.6 The Role of Government

In SusInet, government plays two basic roles: (*i*) to serve as the single trusted third party; (*ii*) to address market and information failures in a similar way as in the SusEco practice.

3.6.1 Single Trusted Third Party

Government plays the role of a single trusted third party in the following way:

Maintain User Identity. The government maintains user identity in the same way as traditional identity information (*e.g.*, driver license, social security number, *etc.*). It should store users' permanent identity and related information in a centralized database, and it is responsible to keep this information confidential. It should provide regular services related to user identity such as registration, reclamation, revocation, *etc.*

Propagate Authoritative Information. The government propagates authoritative information via the *distributed announcement database* service provided by the administration plane. The authoritative information includes: (*i*) *authorization announcements*, *e.g.*, to authorize a company as an agent of the identity authority or as a professional Internet patrol; (*ii*) *authentication information*, *e.g.*, hashed copies of user authentication information (Section 5.1.2), copyright information of data objects (Section 5.2.3); and many others.

3.6.2 Address Market and Information Failures

As pointed out by the authors of [19], the critical role of government in sustainable development is to guide businesses' practices in order to address market and information failures on externalities¹¹. The authors also

pointed out that government should focus on the *market-based approach* to address such failures. In our framework, we directly apply this theory to address businesses' incentives in SusInet.

Here we show an example of SusEco [19] to explain the market-based approach, in particular, *how to provide right incentives and send right signals*. The market-based approach in SusInet is essentially the same. The only difference is the definition of externalities.

The Example: *In the energy industry, the main driver to build new power stations typically comes from the seasonal peak energy demands for cooling and heating. Government's regulatory frameworks which signal the market that "the more energy sold the more money made" could lead to a vicious cycle: The entire system designed to meet peaks carries a redundancy during the predominant non-peak periods; therefore, energy suppliers are motivated to sell the excess capacity to the market, which results in even higher energy demands during peak times. To address this, enlightened regulatory frameworks encourage energy conservation and efficiency. For example, we can introduce higher pricing at peak times to reduce the peak demand. However, in addition to the positive signal sent to users to avoid using energy at critical times, this approach also sends a perverse message to energy suppliers that they can make windfall profits during peak periods. Nevertheless, we can take an alternative approach to solve this. Instead of having the revenue from higher prices at peak times go to energy suppliers, we can have it go into a government fund. This fund can be used as an incentive for the electricity industry in ways that are consistent with long-term societal objectives (*e.g.*, to pursue energy efficiency).*

3.6.3 Unified Incentive System

We alter the market-based approach for businesses slightly and create a version that can be applied to individual users. The central idea of the change is to find counterparts of subsidies and levies¹² that can work for individual users. This new version of market-based approach is the *unified incentive system* (UIS). The term "unified" is used because we bring all Internet services into the same system rather than implementing an incentive system on a per-service basis. This property allows for the maximum efficacy of the incentive system. We propose two versions of UIS.

Basic UIS. First, we introduce a basic version of UIS. We introduce two incentive elements for the system, which are counterparts of subsidies and levies: (*i*) *Reward point*, which is modeled off of the diversified forms of credit point used by businesses that can be redeemed for gifts or money. Reward points are issued by government and are supported by a government fund. The government encourages businesses' SusInet practices by giving both subsidies and reward points. Businesses then use the reward points to encourage good behavior of their customers. (*ii*) *Compliance score*, which is modeled off of the credit score in the credit card system.

¹¹In SusEco, the externalities are the pressure on the natural environment. In SusInet, they are the pressure on a set of culture norms that Internet development relies on.

¹²In the market-based approach, the government usually uses subsidies and levies as an incentive for businesses.

¹⁰There are already such practices in use today, *e.g.*, companies are hired to audit P2P traffic.

The compliance score quantifies the extent to which a user complies with standards. A user can benefit from a high score and can suffer from a low score. For example, if her compliance score is too low, she might be blocked from accessing the Internet.

An Internet user earns reward points and raises her compliance score by behaving well; her compliance score decreases if she violates a standard. The UIS maintains the compliance score and reward points for each user and binds them to her identity. The integrity of the UIS can be enforced through standards and auditing.

Advanced UIS. Basic UIS is the first stage of the UIS. In the long run, the UIS evolves to incorporate advanced features. One advantage feature is to break the compliance score down into different scores that evaluate a user’s behavior from different angles. In addition to compliance, these scores evaluate a user’s contribution and ability as well (*e.g.*, a user’s respect to copyright, a user’s contribution in collaborative work, a user’s creativity, responsibility, *etc.*), which allows for more personalized incentive mechanisms. Indeed, the advanced UIS is a comprehensive *trust system* [5]. Such a trust system is useful not only for Internet development but for the entire economic and social development as well.

3.7 Pyramid Process

In [19], the authors have given a general solution to sustainable development, *i.e.*, the *Pyramid process* (*Pyramid* for short). In this section, we briefly introduce the Pyramid and describe how we apply it to SusInet. In addition, we argue that the Pyramid can work better in SusInet than in SusEco.

3.7.1 The Pyramid

The Pyramid (Figure 3) is a multi-stakeholder¹³ engagement process that is especially suitable to address the huge challenges of sustainable development. Such challenges come from three aspects: (i) Sustainable development requires contributions from all parties — governments, business, civil societies and international bodies. (ii) It creates unprecedented demands for learning, thinking, planning and decision-making. (iii) Initiatives seeking to promote sustainability are often doing so under a sense of time urgency, with limited resources; we do not have time or money to waste on suboptimal solutions or difficult-to-achieve agreements.

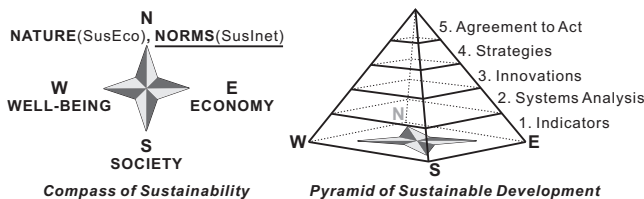


Figure 3: Pyramid Process

At its core, the Pyramid is a framework and a process for strategic planning. However, it can also be used as a training program. The Pyramid incorporates two ba-

sic components: (i) *Compass of sustainability*, a way of representing the different dimensions of sustainability, and of supporting true multi-stakeholder engagement acts as the base of the Pyramid. In SusEco context, the Compass includes the following four points: *nature, economy, society, and well-being*¹⁴. Sustainable development must improve all four compass points, usually at the same time, and often against a backdrop of one or more critically negative trends. (ii) *ISIS method*, a sequential strategic thinking process that helps groups develop a more systematic and strategic understanding of sustainable development. It includes four steps: *indicators, systems analysis, innovation, and strategy*. When coupled with the Compass, it can support group learning, planning, and decision processes that are (to borrow language from NASA) “faster, better, and cheaper”. In addition, it can produce, as a purposeful by-product, improved levels of interdisciplinary understanding and innovative thinking.

3.7.2 Exploit the Pyramid in SusInet

To apply the Pyramid to SusInet, we only need to make a minor change to the Compass: We replace the north compass point, *Nature*, with *Norms*, which refers to the set of culture norms that SusInet addresses.

The Virtuous Cycle. Due to distinct advantages of communication on the Internet, the Pyramid process can work better in SusInet than in SusEco. Our framework adds controllability to the Internet, through which we can provide a proper platform for the Pyramid process. In addition, SusInet can cultivate a good environment for the Pyramid process, including incentives for collaboration, creativity, high information quality, sense of responsibility, trusts, *etc.* In this way, a virtuous cycle forms between the Pyramid and SusInet.

The Role of the Pyramid. In SusInet, the Pyramid plays two roles. First, it is a general solution to SusInet framework (for which this paper is merely a prototype). Second, in our framework, the Pyramid helps with the enactment of standards. It ensures that new standards reach global agreements among service providers and other entities. Meanwhile, it guarantees that the process meets the responsiveness requirement of standards.

4. EVOLVABILITY

Evolvability is one of the two major components of sustainability. It is a critical aspect of the design of the future Internet. Many research studies on the new Internet architecture (*e.g.*, [13, 27, 30]) provide design guidelines for evolvability. Our framework follows their common design principles. In addition, we add new insights to evolvability both from a high-level perspective and through in-depth consideration.

4.1 Apply Common Design Principles

We follow three common design principles on evolvability: *incremental deployability, evolvability of the new*

¹³Stakeholders are those who have an interest in a particular decision, either as individuals or representatives of a group.

¹⁴Well-being is the satisfaction and happiness of individual people — their health, their primary relationships, and the opportunities they have to develop their full potential.

architecture, and testability of evolution.

Incremental Deployability. We should be able to incrementally move from the current Internet architecture to the new one. In our framework, we add a separate default-off network and connect it to current default-on Internet only at end hosts. We make no changes for the Internet’s existing infrastructure, *e.g.*, no changes to any existing routers. The fundamental controllability of our default-off network only relies on its edge part such that we allow the maximum incremental deployability for the transit part. The transit part can reuse current Internet infrastructure during the early stages of deployment, *e.g.*, the transit part of the data plane can be implemented through overlays and that of the administration plane can be implemented via cryptographically protected tunnels. Later, when the default-off network becomes popular, it can gradually add its own physical infrastructure to optimize service performances and to support new services previously unfeasible on the old infrastructure. The edge part can also be incrementally deployed by gradually adding new edge routers.

Evolvability of the New Architecture. Once deployed, the new architecture must have its own long-term evolvability. In our framework, we address this through two kinds of decoupling: (i) We decouple the default-off network into the transit part and the edge part. The fundamental controllability only relies on the edge part. Therefore, the transit part has the maximum design flexibility to address evolvability. (ii) We decouple the network layer into the variant part and the invariant part. All components subject to evolve are put into the variant part, whose long-term evolvability is under *regular* design consideration (Section 4.2.1).

Testability of Evolution. We must be able to test the new Internet architecture with real traffic during its incremental deployment. To this end, we follow the idea given in [13, 27], that is, to exploit traffic *redirection*. In our framework, we redirect user traffic from the old (default-on) network to the new (default-off) network on a per-service basis. In addition, we take the idea of redirection one step further by addressing incentives for redirection (Section 4.2.2).

4.2 Apply New Insights

4.2.1 Implement the Variant Part

The variant part in the new network layer is subject to evolve. A central criteria to implement the variant part is making it globally upgradable, *e.g.*, all routers can upgrade specific software components consistently. Here we introduce two independent solutions, which can also be applied in combination.

A Perfect Solution. A candidate solution might be adapting the centralized control idea of enterprise networks [15, 34] to the wide area network context. However, this approach can suffer from formidable scalability challenges. Nevertheless, there is another solution that can perfectly address the global upgradability, that is, the “active network” idea proposed in [30]. Instead of having routers passively process packets, the

“active network” makes packets (which the authors call “capsules”) contain customized programs executed at each router they traverse. To apply this solution to our framework, we can have the invariant part of the new network layer contain pre-defined program methods that can be invoked by “capsules” and implement the variant part via “capsules”.

Although this solution is beyond what we can readily apply, it can bring about a fundamental innovation of Internet development. As previously mentioned, after the early stages, our default-off network will gradually expand its own brand new infrastructure. We argue this expansion provides a good chance to develop and test the “active network” and bring it to maturity.

Our Solution. In our framework, we exploit the high level controllability to solve the global upgradability issue. We set *standards* that are globally agreed upon by service providers through the Pyramid process. Standards ensure consistent upgrading of the network. We leverage *incentives* and *auditing* to enforce standards. The unique feature of our solution is not to solely depend on technical means³. Consequently, our solution can contribute not only to Internet development, but also to the entire economic and social development.

4.2.2 Implement Redirection

As pointed out in [13, 27], traffic redirection plays a critical role in evolvability. However, there are two questions related to the implementation of redirection.

The first question is *who do the redirection?* There are generally two options: (i) Users choose to redirect their service traffic to the new network. (ii) ISPs redirect user traffic. We argue that the former is the right choice, because only by this choice can we ensure the *robustness* of the new network. This is because user redirection is an effective feedback mechanism for performance of the new network. If not satisfied, a user can choose to redirect her traffic back to the old network.

Then the second question comes: *what are the incentives for a user to redirect?* If performance of a service on the new network is better than it used to be, then it is natural for users to choose to redirect. However, what if, *from user’s perspective*, the service on the new network shows almost the same or even slightly worse performance than on the old network? It seems users have no incentives to redirect in this case. Our framework can effectively address this issue by leveraging the unified incentive system. We can promote a service by giving users benefits for other services.

Therefore, we implement redirection on a per-service basis through the following steps: (i) Motivate users to redirect a service (*e.g.*, P2P content delivery) to the new network. (ii) Test and revise the service to make it more and more robust. (iii) The service reaches maturity and the majority of users move to the new network. (iv) Announce the shutdown of the service on the old network, give time for the remaining users to redirect. (v) Shut down the service on the old network.

4.2.3 Service Driven Deployment

One critical component of our default-off network may not be incrementally deployable — the identity authority, which should be supported by government. To solve this issue, we propose the idea of *service driven deployment*: We select a suitable service as a starting point. By deploying this service, we get the identity authority for free, *i.e.*, as a by-product. We require this first service to be extremely useful such that it is worth what it costs to deploy. Meanwhile, it should provide sufficient motivation for government to join.

There are currently two candidates of this first service in our mind: (i) A common platform for *real name systems*. A real name system has the following properties: one account per user, user identity is traceable by authority, partial user information is resolvable. Meanwhile, it should also retain a user’s privacy. Real name systems are very useful for businesses. (ii) An advanced idea sharing platform¹⁵. To start, this platform may only connect a few academic institutes and large businesses for the Pyramid process. Later, the platform could evolve to a common idea sharing platform or a general purpose information retrieval, learning, and decision platform.

5. IMPLEMENTATION EXAMPLES

In this section, we provide implementation examples to concretely show ideas in our framework.

5.1 Identity and Identity Authentication

5.1.1 Identity

The following is an implementation example of user identity that follows the design criteria in Section 3.4.1.

PID and TID. The identity authority issue each user a *permanent identity (PID)* through registration. Along with the *PID*, the identity authority also issues a number of secret codes (*SEC*) to the user, each *SEC* is corresponding to a day. Both the *PID* and the *SEC* are stored on a small hardware device — *network passport (N-pass)*. To access the Internet, a user plugs her N-pass into the computer. The N-pass helps her computer to generate *temporary identities (TID)*, which are used for authentication. The relationship between *TID* and *PID* can be formulated in the following way:

$$\begin{cases} TID = f(PID, time, seeds, PubKey) \\ PID = g(TID, time, seeds, PriKey) \end{cases} \quad (1)$$

PubKey, PriKey: the public, private key pair of the identity authority. *f*: a function to generate *TID* from *PID*. *g*: a function to resolve *PID* from *TID*. *seeds*: any commonly known parameters (*e.g.*, service type)

Such an implementation makes the observed identity, *TID*, be *untraceable* by unprivileged users while resolvable by the identity authority.

Authentication. To perform authentication, we require the user to supply a verification code (*VC*) along with *TID*. The *VC* is computed based on the user’s

SEC, which is only known by the user herself and the identity authority. The formula to generate *VC* is:

$$VC = v(TID, SEC) \quad (2)$$

v: a commonly known hash function.

Upon receiving the *TID* and *VC*, the identity authority first resolves the user’s *PID*, which in turn helps to retrieve the user’s *SEC*. Then it verifies the *VC* by regenerating it the same way as the owner does.

Address Identity Theft. To address identity theft, we can embed a clock into the N-pass. Using the clock, we can enforce that the stored *SEC* for a specific day can be accessed *only* on that day (*i.e.*, impossible to access in advance). This effectively reduces the risk of leaking *SECs*. Still, chances are that the N-pass itself could be stolen. To solve this, the identity owner can reclaim her N-pass via the identity authority by changing her *SECs*. A user can easily detect identity theft by looking at her compliance score in the UIS (if someone else uses her identity to perform misbehavior). Moreover, a decreasing compliance score motivates the owner to quickly respond to the identity theft.

5.1.2 Identity Authentication

Large-scale Replication. A central design criteria of identity authentication is making it scalable. We address this through large-scale replication of the identity authority’s database¹⁶. We call each entity that maintains a replicated database an *agent*. We transfer to each agent the following data: *PID* and *SEC* of all users, the private key (*PriKey*) of the identity authority. However, since this information is confidential, we must address the security issue. To do so, we can exploit cryptographic hash functions. We replicate a hashed copy instead of the original data.

First, the identity authority generates a public and private key pair (*PubKey_i, PriKey_i*) for each agent, *i*. Second, we modify Formulas (1) and (2) in the previous section to Formulas (3) and (4) respectively:

$$\begin{cases} TID = f(PID, time, seeds, PubKey_i) \\ PID_i = g(TID, time, seeds, G(PriKey_i)) \end{cases} \quad (3)$$

$$\begin{cases} VC = v(TID, SEC) \\ V_i(VC) = v(TID, V_i(SEC)) \end{cases} \quad (4)$$

G, V_i: cryptographic hash functions.

Third, we disseminate the following data to agent *i*: *PID_i* and *V_i(SEC)* of all users, *G(PriKey_i)*, and *V_i*.

When a user generates a *TID* for authentication, she uses the public key of an agent (*PubKey_i*) instead of that of the identity authority (*PubKey*). Each agent stores a hashed copy of user identity (*PID_i*) and it is resolvable from the *TID* by using a hashed version of private key (*G(PriKey_i)*). The way a user generates a verification code (*VC*) remains the same (first line of

¹⁶Given that this database is *read-only* (only writable by identity authority and writing is infrequent) and its data volume is *bounded* (one identity per user), we can implement the large-scale replication in a simple and efficient way.

¹⁵Creating an environment that allows higher level of idea sharing and creativity is a fundamental innovation [23].

Formula (4)). Each agent stores a hashed copy of secret codes ($V_i(SEC)$), based on which it can check the validity of verification codes (using the second line of Formula (4)), thereby authenticating the user. In this way, we achieve to replicate authentication information without compromising security. Authentication information stored by one agent is useless to other agents (can neither resolve nor distinguish user identities).

*Announce Public Keys*¹⁷. One subtlety in the above implementation is: *How do we announce the public key of each agent ($PubKey_i$) such that users can verify its validity?* We solve this issue through digital signatures. The identity authority signs each announced $PubKey_i$ using its private key. In addition, it announces a blacklist of revoked public keys with each entry signed as well. Users are responsible for keeping their blacklists up-to-date. To check the validity of a public key, a user not only checks the signature of an announced public key, but also refers to the blacklist.

5.2 Service Example: Copyright Protection

In this section, we give a service example, copyright protection, to show how to leverage *identity*, *standards*, *incentives*, and *auditing* in practice.

The goal of this copyright protection service is to form a virtuous cycle: (i) Users respect the copyright owners by paying a reasonable copyright fee; with a large number of compliant users, the price of copyrighted digital products can be significantly reduced. (ii) Due to increased income from copyright fees, people are motivated to generate higher-quality products, and sellers are motivated to provide better sales platforms and better value-added services. (iii) Users benefit from downloading high-quality digital products and from the convenient ways to find what they really want, are therefore willing to pay the relatively small copyright fee.

5.2.1 Incentives, Standards, and Prevention

First, we exploit the UIS to honor users who respect copyrights. For example, whenever a user purchases a copyrighted product (*e.g.*, buy software or an e-book online, purchase a movie at the local DVD store, *etc.*), we give her credit by either offering reward points or raising her compliance score via the UIS.

Second, we set a standard for *content delivery services*¹⁸: For a content delivery service, the source endpoint must provide the data identity in the network layer metadata for the data object that she wants to send during the session setup phase; both endpoints are required to verify the consistency (Section 3.4.2) between the data identity and the actual data object; if inconsistent, the source endpoint should not deliver the object, and the destination endpoint should drop it.

The data identity encodes the identity of the user (resolvable by the identity authority) who created the

data object. Therefore, it poses a threat to anyone who wants to generate copyright infringing products, *e.g.*, “cracked” software, because she must use her own user identity to generate the data identity.

5.2.2 Inspiring Learning

Given the above threat, users are not likely to deliberately violate copyright laws. However, they can still make unintentional violations due to the extreme complexity of copyright laws. For example, they could misinterpret “fair use” or fail to understand whether a given digital item is copyrighted or not.

The user identity encoded within the data identity allows us to design approaches that can inspire users to learn about copyright-related issues both passively and actively. For example, when a copyright infringing object is detected, we can inform the user who created the object and let her know the related copyright information. This is passive learning. On the other hand, we can apply moderate penalties (*e.g.*, to deduct her compliance score) for unintentional copyright violations. This provides an additional incentive for her to actively learn about copyright laws, such that she can avoid further penalties in the future.

5.2.3 Authenticating Data Identity

Copyrighted products are registered in a database managed by a government body — *copyright authority*. When a content delivery service begins, the source edge router interacts with an administration plane interface provided by the copyright authority for data identity authentication.

Blocking Unauthorized Distribution. The copyright authority first checks whether the data identity maps to a registered object in its database. If it does, the authority verifies whether the session associates with a valid copyright authorization, *e.g.*, whether the destination user owns an authorization for downloading or whether the source user owns an authorization for distributing the object. Based on the authentication result, the edge router either established or denies the session. In this way, unauthorized distribution of *registered* data objects is blocked.

Blocking a Copyright Infringing Object. The above method is incapable of blocking *copyright infringing objects*, which are not registered in the database. What should happen if a copyright infringing object has been spread on the Internet¹⁹? To address this, the copyright authority manages a blacklist in its database. When a copyright infringing object is detected, it is added to the blacklist such that further delivery attempts of this object can no longer pass the data identity authentication, hence, it can no longer spread over the Internet.

5.2.4 Public Auditing

How do we detect copyright infringing objects that

¹⁷This mechanism is also the key to implementing the *distributed announcement database* of the administration plane.

¹⁸We define a content delivery service as any transfer of copyrighted data; the transfer could be in the client-server mode, P2P file sharing mode, or as an email attachment, *etc.*

¹⁹Although the existence of the data identity can effectively prevent copyright violations, the distribution of a copyright infringing object may still happen commonly, *e.g.*, due to malfunctioning software or unintentional violations.

are not yet blacklisted? Despite the fact that computers can certainly help, identifying such objects is still far beyond what computers can do nowadays. As a result, human efforts must be involved.

Public Patrol. The copyright authority can authorize professional companies to be the Internet patrol, who detect copyright infringing objects and blacklist them. In addition to these professional patrols, we can also have *public patrols*. Public patrols are formed by unprivileged Internet users who report copyright violations, and necessarily get rewards for doing so. Public patrols can help to significantly reduce the professional patrols' workload. As a result, professional patrols can focus on verifying objects commonly reported by public patrols and adding verified infringing objects to the blacklist. Although the public patrol approach appears simple, its implementation is non-trivial. This is because it relies on a comprehensive incentive mechanism; otherwise, a number of social and economic problems could arise. To implement it, we can exploit the UIS, which can effectively address incentives for individuals.

Public Patrols and Auditing. Public patrols also help in auditing. For example, (i) audit source edge routers by verifying that they correctly block unauthorized distribution and copyright infringing objects; (ii) audit source endpoints (public content delivery servers in particular) by verifying that a source endpoint guarantees the consistency between the data identity and the data object actually sent. Although public patrols might not be able to provide irrefutable evidence, they can always help provide abundant clues to facilitate auditing.

5.3 Miscellaneous Service Examples

Table 1 summarizes 7 miscellaneous service examples. In this table, we focus on the question of what new features of our framework (compared with current Internet practice) help to satisfy demands of new services and solve problems of the current Internet.

6. RELATED WORK

Our research learns and applies ideas from numerous related work across diversified topics, including: capability (*e.g.*, [8, 26, 31, 36]), game theory based behavioral study and solution (*e.g.*, [6, 25, 29]), centralized administration (*e.g.*, [15, 34]), reputation systems (*e.g.*, [5, 32]), incentives (*e.g.*, [18]), information privacy and trust (*e.g.*, [3, 17]), auditing (*e.g.*, [9, 20, 28]), new Internet architecture (*e.g.*, [13, 27, 30]), new host identity and address schemes (*e.g.*, [7, 22]), cryptographic tools (*e.g.*, [14, 16]), network management and fault diagnosis (*e.g.*, [10, 12]), *etc.* Due to space limit we select representative ones and disseminate their main ideas in proper parts across the entire paper.

In addition, as the interdisciplinary research, SusInet also integrates ideas and principles of economics, sociology and law (*e.g.*, [19, 23, 24]).

7. CONCLUSIONS

In this paper, we propose a solution to the fundamental problem of current Internet, *i.e.*, insufficient sustainability. We address sustainability by adding proper *controllability* and a *semantics aware property* to the architecture and by introducing four critical leverage points: *identity, standards, incentives, and auditing*. We learn and apply basic ideas from traditional sustainable development and find they are well suited for sustainable development of the Internet. Most surprisingly, we find they work better in this new context. Sustainable development of the Internet can lead to fundamental innovations not only for the Internet but for scopes beyond it (entire economic and social development) as well.

8. REFERENCES

- [1] The FIND project. <http://www.nets-find.net/>.
- [2] The GENI project. <http://www.geni.net/>.
- [3] LOAF. <http://loaf.cantbedone.org/>.
- [4] E. Aboujaoude, L. Koran, N. Gamel, M. Large, and R. Serpe. Potential markers for problematic Internet use: a telephone survey of 2,513 adults. *CNS Spectrums: The International Journal of Neuropsychiatric Medicine*, 2006.
- [5] T. B. Adler and L. de Alfaro. A content-driven reputation system for the wikipedia. In *ACM WWW*, 2007.
- [6] A. Akella, S. Seshan, R. M. Karp, S. Shenker, and C. H. Papadimitriou. Selfish behavior and stability of the Internet: a game-theoretic analysis of TCP. In *SIGCOMM*, 2002.
- [7] D. Andersen, H. Balakrishnan, N. Feamster, T. Koponen, D. Moon, and S. Shenker. Holding the Internet accountable. In *Hotnets-VI*, 2007.
- [8] K. Argyraki and D. Cheron. Network capabilities: The good, the bad and the ugly. In *HotNets-IV*, 2005.
- [9] K. Argyraki, P. Maniatis, O. Irzak, S. Ashish, and S. Shenker. Loss and delay accountability for the Internet. In *ICNP*, 2007.
- [10] P. V. Bahl, R. Chandra, A. Greenberg, S. Kandula, D. Maltz, and M. Zhang. Towards highly reliable enterprise network services via inference of multi-level dependencies. In *SIGCOMM*, 2007.
- [11] H. Ballani, Y. Chawathe, S. Ratnasamy, T. Roscoe, and S. Shenker. Off by default! In *HotNets-IV*, 2005.
- [12] H. Ballani and P. Francis. Conman: A step towards network manageability. In *SIGCOMM*, 2007.
- [13] A. Bavier, N. Feamster, M. Huang, L. Peterson, and J. Rexford. In VINI veritas: realistic and controlled network experimentation. In *SIGCOMM*, 2006.
- [14] M. Bellare, D. Micciancio, and B. Warinschi. Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions. In *EUROCRYPT*, 2003.
- [15] M. Casado, M. J. Freedman, J. Pettit, J. Luo, N. McKeown, and S. Shenker. Ethane: taking control of the enterprise. In *SIGCOMM*, 2007.
- [16] D. Chaum and E. van Heyst. Group signatures. In *EUROCRYPT*, 1991.
- [17] P. A. Dinda. Addressing the trust asymmetry problem in grid computing with encrypted computation. In *ACM LCR*, 2004.
- [18] J. R. Douceur and T. Moscibroda. Lottery trees: motivational deployment of networked systems. In *SIGCOMM*, 2007.
- [19] K. C. H. *et al.* *The Natural Advantage of Nations (Vol. I): Business Opportunities, Innovation and*

²⁰The previous two waves are *mainframe* (one computer, many people) and *PC* (one computer, one person).

Demands / Problems	Solution in SusInet Framework
1. Quality of Service (QoS). QoS such as low delay is almost impossible to achieve through overlay approaches (<i>e.g.</i> , overlay routing, tunneling). Other QoS such as bandwidth or reliability guarantees will be much easier to achieve at lower layers than through overlay approaches. Although there exist practical <i>intra-domain</i> QoS solutions, the current practice of <i>inter-domain</i> routing impedes global cooperation among ISPs to support QoS extension.	The new network layer is <i>semantics aware</i> . It can support more diversified transportation services than current Internet does (only two services: datagram (UDP) and stream (TCP)). It provides good <i>downward extensibility</i> , which allows lower layer infrastructures to continuously upgrade and allows multiple routing architectures to work in parallel. In addition, <i>cooperation</i> among ISPs is no longer an intractable issue in our framework (Section 4.2.1).
2. Why IP Option Does Not Work? The IP option field is designed to provide additional flexibility for the IP protocol. It seems to be the equivalent of the network layer <i>metadata</i> that we propose. But in practice it is rarely used and helps little for Internet evolvability due to both processing efficiency concerns and impediments in the global enforcement of new options.	The network layer metadata can be transferred through an <i>out-of-band</i> control channel. It provides advanced flexibility by supporting <i>long</i> control information. <i>Decoupling</i> the network into the edge and transit parts improves processing efficiency for metadata. Intensive processing can be restricted to the edge. In addition, <i>global enforcement</i> (Section 4.2.1) is no longer a big problem in our framework.
3. Network Management and Fault Diagnosis. As pointed out in [10,12,22]: (i) The difficulty of network management today is partially attributed to too many details exposed by heterogeneous infrastructures and diversified protocols at different layers. (ii) Fault diagnosis can benefit a lot from correlating diversified elements at different layers.	We introduce the <i>Network-S</i> sublayer to address this. This sublayer serves as a <i>common interface</i> for management and fault diagnosis functionalities of all layers. It provides a high-level <i>abstraction</i> to significantly reduce complexities resulting from the ever-evolving data plane. In addition, this sublayer has good <i>evolvability</i> .
4. Immunity. We need more comprehensive solutions to counter virus, malware, spam, <i>etc.</i> In certain cases, we may even want to enforce the immunity globally.	We can exploit ideas similar to our copyright protection solution. The use of <i>data identity</i> can effectively deter generation and dissemination of virus, malware, and spam.
5. Collaborative Editing. The idea of collaborative editing (<i>e.g.</i> , Wikipedia) can become much more useful than today's practice if <i>comprehensive controllability</i> is available, <i>e.g.</i> , effective approaches to counter vandalism.	SusInet provides a straightforward solution to the comprehensive controllability needed. Furthermore, it provides not only solution to controllability but also solution to <i>incentives</i> that motivate users to contribute.
6. Real Name System and Anti-Addiction. For management purposes (<i>e.g.</i> , Internet addiction recovery, improvement of users' responsibility for their posts on Internet forums), sometimes we prefer to use a real name system — one account per user, user identity traceable by authority, partial user information (<i>e.g.</i> , age, gender) resolvable. Meanwhile, we should also protect users' privacy.	The identity authority and its agents can provide <i>hashed version of permanent user identities</i> to entities (businesses or organizations) who need real name systems. Each entity is given a different hash function or key so that its data is useless elsewhere. The identity authority can also expose partial user information to authorized entities. Real user identity is resolvable only by the identity authority.
7. Ubiquitous Computing. Ubiquitous computing (one person, many computers) [33] names the third wave ²⁰ in computing. It was first articulated in 1988, but has not yet prevailed as some people once predicted.	SusInet progress can meet the demanding requirements of ubiquitous computing, including: reliable online storage services (enforced through auditing [28]), user and data privacy protection, reliable Internet access, <i>etc.</i>

Table 1: Miscellaneous Service Examples

- Governance in the 21st Century*. Earthscan/James & James, 2005.
- [20] A. Haeberlen, P. Kuznetsov, and P. Druschel. PeerReview: Practical accountability for distributed systems. In *SOSP*, 2007.
- [21] G. Hardin. The tragedy of the commons. *Science*, 1968.
- [22] M. Karsten, S. Keshav, S. Prasad, and M. Beg. An axiomatic basis for communication. In *SIGCOMM*, 2007.
- [23] L. Lessig. *The Future of Ideas: The Fate of the Commons in a Connected World*. Random House Inc., 2002.
- [24] L. Lessig and L. Lessig. *Code and Other Laws of Cyberspace*. Basic Books, Inc., 2000.
- [25] R. Mahajan, D. Wetherall, and T. E. Anderson. Mutually controlled routing with independent ISPs. In *NSDI*, 2007.
- [26] B. Parno, D. Wendlandt, E. Shi, A. Perrig, B. Maggs, and Y.-C. Hu. Portcullis: Protecting connection setup from denial-of-capability attacks. In *SIGCOMM*, 2007.
- [27] S. Ratnasamy, S. Shenker, and S. McCanne. Towards an evolvable Internet architecture. In *SIGCOMM*, 2005.
- [28] M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan. Auditing to keep online storage services honest. In *HotOS-XI*, 2007.
- [29] G. Shrimali, A. Akella, and A. Mutapcic. Cooperative inter-domain traffic engineering using nash bargaining and decomposition. In *INFOCOM*, 2007.
- [30] D. L. Tennenhouse and D. J. Wetherall. Towards an active network architecture. *SIGCOMM Comp. Comm. Rev.*, 2007.
- [31] M. Walfish, M. Vutukuru, H. Balakrishnan, D. Karger, and S. Shenker. DDoS defense by offense. In *SIGCOMM*, 2006.
- [32] K. Walsh and E. G. Sirer. Experience with an object reputation system for peer-to-peer filesharing. In *NSDI*, 2006.
- [33] M. Weiser. The computer for the twenty-first century. *Scientific American*, 1991.
- [34] H. Yan, D. A. Maltz, T. S. E. Ng, H. Gogineni, H. Zhang, and Z. Cai. Tesseract: A 4D network control plane. In *NSDI*, 2007.
- [35] X. Yang, D. Clark, and A. W. Berger. NIRA: a new inter-domain routing architecture. *IEEE/ACM ToN*, 2007.
- [36] X. Yang, D. Wetherall, and T. Anderson. A DoS-limiting network architecture. In *SIGCOMM*, 2005.