# Standards And Sustainable Infrastructures: Matching Compatibility Strategies With System Flexibility Objectives

**By Tineke M. Egyedi[20]**

*Problems of entrenchment often severely hamper the introduction of change in large technical systems (LTSs). They lack the flexibility to innovate.*

*This paper explores the counter-intuitive assumption that standards increase system flexibility. To what degree and in what manner can standards—and other strategies that create technical compatibility—enhance system flexibility? It focuses on information networks of which the life cycle is sometimes needlessly short. Different objectives of system flexibility can be discerned (e.g., exchangeability and longevity). I examine to what degree specific compatibility strategies (i.e., gateway technologies, standardisation, modularity and interactive compatibility) can be matched with distinct flexibility objectives.*

*I conclude that compatibility is crucial to sustainable system innovation, and recommend that innovation policies should incorporate standards policy.*

20 A version of this article was published earlier in Interoperable Nederland, Nico Westpalm van Hoorn, Peter Waters en Pieter Wisse (Eds.). Den Haag: Forum Standaardisatie, 2011, pp.379-391; and Unifier or Divider? Sherrie Bolin (Ed.), USA, Canton: Bolin Communications, Standards Edge Series, 2010, pp. 223-234.

# Entrenchment

There is no discussion about the need to make large technical systems (LTSs) such as transport and energy systems more sustainable environmentally, economically, and socially. However, many LTSs seem impervious to change. This is partly due to the number, interrelatedness and interdependence of constituent socio-technical components and subsystems. LTSs comprise technical artefacts as well as institutional and regulatory elements of artefact production and use. Organizations and companies develop and sustain the system. Technical add-ons and complementary products are created. As an LTS expands, the number of and interdependencies between actors and artefacts grows. Over time, these interdependencies crystallise, solidify, and make manifest a process of socio-technical *entrenchment.*[21] To paraphrase Collingridge, changes are only possible at the expense of readjusting the technologies and other socio-technical arrangements that surround them. The larger the vested interests, the higher the costs of change.

**Box 1: Entrenchment in ICT**

In a large government agency, the ICT infrastructure had evolved in a piecemeal fashion. Bit by bit stand-alone, local provisions were coupled and integrated with networked functionalities. Of the 350 software systems, 150 were generic and used throughout the organisation (e.g., office software). Two hundred software systems served a special purpose and were used by specific people or only locally. Those involved identified a number of serious problems with respect to system maintenance and evolution:

- the short life cycle of IT products. IT products have a relatively short lifecycle. The average time for a software upgrade is about three years. This

[21] David Collingridge, *The Social Control of Technology* (Milton Keynes, UK: The Open University Press, 1981), 47.

period is close to the time needed to roll out IT products in a large organisation (i.e., from idea to working implementation). As a result, there is a continuous pressure to upgrade the infrastructure.

- different local needs. Different IT configurations at the local level (i.e., lower level of organisational unit) make it difficult to rollout IT products organisation-wide. Locally adaptations are introduced that further increases the differences between local configurations.
- unsustainable software design. Too little attention is paid to sustainable software design. For example, software developed in a certain programming environment does not automatically run in another (user) environment.
- unexpected interaction between software**.** New applications sometimes affect existing ones in unexpected ways.
- provider dependence. The organisation is sometimes locked into provider-dependent (closed source) software, such as off-the-shelf software of a monopolist and tailor-made software. System maintenance can become very dear.

The case illustrates that where ". . .  information systems are updated, . . . frequently, the resulting system grows increasingly complex, as does the maintenance process itself . . ."[22] The complexity and further development of the ICT infrastructure become difficult to manage. The ICT system lacks the necessary flexibility.

An example of undesirable entrenchment is the production of polyvinyl chloride (PVC).[23] From the early 1930s onwards, its production posed health and environmental risks. The dangers ranged from health risks to workers and those living near

---

[22] Nancy Bogucki Duncan, "Capturing Flexibility of Information Technology Infrastructure: A Study of Resource Characteristics and
their Measure," Journal of Management Information Systems, 12, no. 2 (Fall 1995):  43.

[23] Karel Mulder and Marjolijn Knot, "PVC Plastic: A History of Systems Development and Entrenchment," Technology in Society 23, no. 2 (2001): 265-286.

production and processing plants (toxicity and carcinogenity caused by vinyl chlorine; Miamata disease due to mercury emission) to the dioxin found in cow milk as a result of incineration of PVC waste in the 1980s. Despite public protests, PVC is still produced nowadays. While the industry has improved its production, ironically this has reinforced PVC entrenchment, making the industry's conversion to non-chlorinated plastics less likely.

Apparently, such large technical systems have a *technological momentum*[24] that ". . . pushes the system along a path-dependent process of technological change . . . ".[25] Unless something radical happens, no real deviation from the set path will occur.

Theoretical concepts such as technological momentum and path-dependency suggest that significant system changes are unlikely. They reflect a deterministic view on LTS evolution and provide few clues for policy intervention. The corresponding policy dilemma, the Collingridge Dilemma, is that entrenchment problems are difficult to foresee at an early stage of technology development and are difficult to address at a later stage. Where infrastructure change is aimed for, other concepts are more promising. For example, under the heading of 'de-entrenchment strategies', Mulder and Knot propose means to recreate the critical space necessary for system change. These strategies target the system’s actor network by negotiating about and redefining aspects of the critical problem (e.g., solving a different problem or

---

[24] Thomas P. Hughes, “The Evolution of Large Technological Systems,” in *The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology*, ed. Wiebe E. Bijker, Thomas P. Hughes, and Trevor J. Pinch (Cambridge, MA: MIT Press, 1987), 51-82.

[25] Andrew Davies, “Innovation in Large Technical Systems: The Case of Telecommunications,” *Industrial and Corporate Change* 5, no. 4 (1996): 1148.

assigning a new problem owner; giving in to demands with regard to one part of the LTS in order to safeguard another; and defining the problem at a higher level in order to avoid competition within the actor network at a lower level).

In this article I focus on ways to enhance the flexibility of LTSs. Paraphrasing Feitelson and Salomon,[26] flexibility refers to the ease with which an LTS can adjust to changing circumstances and demands. It is about openness to change. Thus, a flexible design would make a system less susceptible to unwelcome, premature entrenchment. In particular, I look at standards as a means to enhance system flexibility. Certain authors note that *compatibility* or interoperability standards play a crucial role in the evolution of LTSs, but few discuss how they actually relate.

## Paradox Of Standards

There is an intuitive tension between standards and flexibility.[27] Standards may foremost seem catalysts of entrenchment for two related reasons. First, standards codify existing knowledge and practices. In Reddy's wordings ". . . standardization . . . is an attempt to establish what is known,

---

[26] Eran Feitelson and Ilan Salomon, "The Implications of Differential Network Flexibility for Spatial Structures," Transportation Research Part A, 34 (2000): 463.

[27] Ole Hanseth, Eric Monteiro, and Morten Hatling, "Developing Information Infrastructure: The Tension between Standardization and Flexibility," *Science, Technologies and Human Values* 21, no. 4 (1996): 407-426.

consolidate what is common, and formalise what is agreed upon."[28] Codification is a primary source of entrenchment.

Second, the interrelatedness of multiple LTS components is a source of entrenchment as well. These components are complementarities.[29] Often, their interfaces are defined by standards. An example is the A4 paper format that specifies the interface between divers paper processing machines (e.g., copying machines, telefaxes and printers) and office requisites (e.g., folders, computer software). The standardised interface eases the entry of new market players, and increases interdependencies between actors and artefacts.[30] It stabilises the market. Entrenchment eventually befalls all useful standards.

However, a standard can also be a means to postpone system entrenchment as standardization in one part of the system creates flexibility in another[31]. Formulated differently, "interdependence among the development of complementary technologies may require the coordination provided by standardization in one

[28] N. M. Reddy, "Product of Self-Regulation: A Paradox of Technology Policy," *Technological Forecasting and Social Change* 38 (1990): 59.

[29] Paul A. David and Shane Greenstein, "The Economics of Compatibility Standards: An Introduction to Recent Research," *Economics of Innovation and New Technologies* 1 (1990): 7.

[30] Carl Cargill, *Information Technology Standardization: Theory, Process and Organizations* (Cambridge, MA: Digital Press, 1989); Reddy, "Product of Self-Regulation," 56.

[31] Geoff J. Mulgan, *Communication and Control: Networks and the New Economies of Communication* (New York: Guilford Press, 1990), 202.

domain so as to foster the generation of diversity in another."[32] For example, the international standard for freight container dimensions (ISO/R 668) lies at the basis of intermodal transport between sea, rail and road transport[33]. It illustrates that standards can also enhance flexibility in LTS design.

In the following the apparently paradoxical role of standards is examined more closely, whether formal standards[34], consortium or de facto standards. It is discussed in the wider context of creating local compatibility without the overall system losing the ability to evolve and innovate.

This paper is structured as follows. First, reasons to strive for system flexibility systems are discussed. Next, issues of compatibility are turned to. Different sources of compatibility and compatibility dimensions are identified. Building on the previous sections, a conceptual model is drawn up that integrates flexibility objectives and compatibility strategies, and carefully distinguish between means and aims. The concluding section readdresses compatibility issues in the light of sustainable system evolution and innovation.

---

[32] Paul A. David, *Standardization Policies for Network Technologies: The flux between Freedom and Order Revisited* (ENCIP Working Paper Series, Montpellier, France: EEIG/ENCIP, October 1994), 25.

[33] Tineke M. Egyedi, "The Standardized Container: Gateway Technologies in Cargo Transport," in *EURAS Yearbook of Standardization*, Vol. 3 Homo Oeconomicus XVII(3), ed. Manfred Holler and Esko Niskanen (Munich: Accedo, 2000), 231-262.

[34] Formal standards are " . . . provisions for common and repeated use, aimed at the achievement of the optimum degree of order in a given context." *ISO/IEC Guide 2: General Terms and Their Definitions Concerning Standardization and Related Activities*, 1991.

# Objectives Of Flexibility

Flexibility is a means and not an end in itself. Therefore, we need to know why system flexibility is sought (i.e., flexibility objectives). Many areas of technology, divers as they may be, share the same objectives.[35] For example, the automobile industry and information managers seek system flexibility to allow the introduction of changes while simultaneously preserving earlier investments. In the automobile industry, flexibility serves the purpose of creating a wider variety of personalised products, however, the general aim is the same as in others areas: to reduce engineering efforts and facilitate system maintenance. Table 1 lists some general, partly overlapping, flexibility objectives.

| **General Flexibility Objectives** |
|---|
| • improvement while preserving earlier investments<br>• reduced engineering efforts<br>• reduced operational costs<br>• higher system efficiency<br>• reduced maintenance efforts<br>• reusability |

Table 1: General flexibility objectives.

[35] Duncan, "Capturing Flexibility of Information Technology Infrastructure"; Takahiro Fujimoto and Daniel Raff, "Conclusion," in *Coping with Variety: Flexible Productive Systems for Product Variety in the Auto Industry*, eds. Yannick Lung, J. J. Chanaron, Takahiro Fujimoto, and Daniel Raff (Aldershot, UK: Ashgate, 1999), 393-406; Feitelson and Salomon, "The Implications of Differential Network Flexibility"; Terry Anthony Byrd and Douglas E. Turner, "An Exploratory Examination of the Relationship between Flexible IT Infrastructure and Competitive Advantage" *Information and Management* 39, no. 1 (November 2001): 41-52.

Looking in more detail into flexibility requirements in ICT, reusability of system components plays a key role. It is relevant to system innovation, reengineering, and managing the rapid change of technological generations. Independent and reusable data and application components simplify ". . . processes of development, maintenance or reengineering of direct-purpose systems," and reduce their costs.[36] Reusability is an overarching aim. It comes in different shapes, and is an important element in many of the following, more specific flexibility objectives in ICT:[37]

- exchangeability, that is, exchangeable software applications, computer hardware, etc. (i.e., reuse in a different system or context, and over time),[38]
- portability, which refers to the different hardware and software platforms upon which software entities can operate and be ported (i.e., reuse on different platforms),[39]
- scalability, which refers to the possibility to use the same software on mainframe and micro-computers (i.e., reuse in smaller/larger system), [40]

---

[36] Ibid.

[37] Tineke M. Egyedi, "Standards Enhance Flexibility? Mapping Compatibility Strategies onto Flexibility Objectives" (paper presented at EASST 2002 Conference, Standardization Track, University of York, UK, July 31-August 3, 2002).

[38] J. A. Dinklo, "Open Systemen," *Informatie en Informatiebeleid* 7, no. 2 (1989): 29-36.

[39] Ibid.

[40] Ibid.

- extendibility or upgradeability (i.e., add new elements to a system in order to reuse existing parts of the system and lengthen its life-span ),[41]
- integration of heterogeneous components and subsystems (i.e., reuse of part of the system by integrating new elements or by integrating different subsystems; organization internally-oriented), [42]
- interconnectivity (i.e., reuse of system through coupling with other (sub)systems; organization externally-oriented),[43]
- reversibility (i.e., reversing changes to the system), and
- downgradeability (likewise, e.g., for accessing an old archive; longevity)

Some flexibility objectives are more likely to be achieved by standards; important; for others, other means of creating compatibility may be more obvious.

[41] Duncan, "Capturing Flexibility of Information Technology Infrastructure."

[42] Reuse of part of a system for the purpose of integration with another system (part) is a transient form of flexibility: once integrated into a higher level system, flexibility is lost at the lower level. An example of integration can be found in Philipp Genschel, "Institutioneller Wandel in der Standardisierung van Informationstechnik" (doctoral dissertation, University of Cologne, Germany, 1993).

[43] Philipp Genschel, "Institutioneller Wandel in der Standardisierung van Informationstechnik" (doctoral dissertation, University of Cologne, Germany, 1993).

# Compatibility

The term compatibility refers to the "suitability of products, processes or services for use together . . ."[44] It is used here as synonymous with 'interoperability'. As a stepping stone towards a discussion of compatibility strategies, I first address key characteristics and possible sources of compatibility.

## Generic and Dedicated Gateways

The term 'compatibility' is closely related to the term 'gateway technology', which refers to " . . . a means (a device or convention) for effectuating whatever technical connections between distinct production sub-systems are required in order for them to be utilised in conjunction, within a larger integrated . . . system."[45] Gateways "make it technically feasible to utilise two or more components/ subsystems as compatible complements or compatible substitutes in an integrated system of production."[46]

Gateways differ in the scope of compatibility they achieve.[47] Some gateways are dedicated. They link an exclusive and specified number of subsystems. For example, gateways that link specific proprietary computer networks belong to this category.

---

[44] ISO/IEC, *ISO/IEC Guide, 2.*

[45] Paul A. David and Julie Ann Bunn, "The Economics of Gateway Technologies and Network Evolution: Lessons from Electricity Supply History," *Information Economics and Policy* 3, no. 2 (1988): 170.

[46] Ibid., 172.

[47] Tineke M. Egyedi, "The Standardized Container".

Other gateways have generic properties. Standards developed in committees[48] function as generic gateways. The example of the A4 paper format was mentioned earlier as an interface specification between unspecified and divers storage and processing devices. An even more generic category of standards is the reference model that guides interdependent, complementary standards activities. A well-known one in the field of ICT is the Open Systems Interconnection (OSI) Reference Model[49]. Gateway technologies can thus be categorised as dedicated, generic, or meta-generic, depending on the scope of compatibility concerned.

The degree of standardization to which a gateway is submitted, determines the scope of the gateway solution. Where no standardization has occurred, the connection between subsystems is improvised, at it were. This corresponds to a dedicated gateway. Standardized gateway solutions, which aim at connecting an unspecified number of subsystems, correspond to generic gateways. Gateways that are based on modelled (standardized) solutions, that is, standardization at the level of reference frameworks, embody meta-generic properties. SeeTable 2.

[48] The term "committee standardization" refers here to activities that are exclusively set up to lead to multi-party standards. They take place in formal standards bodies such as ISO, in professional organizations, and other multi-party fora (IEEE, IETF), or in standards consortia (e.g., W3C; i.e., multi-party industry standards fora).

[49] The OSI reference model (ISO 7498 and CCITT X.200) identifies logically separate generic functions in data communication. It depicts these as a set of hierarchically ordered layers, which address areas of standardization.

| Level of Standardization | Scope of Gateway Solution |
|---|---|
| High (modelled) | Meta-generic |
| Medium (standardized) | Generic |
| Low (improvised) | Dedicated |

Table 2: Relationship between the level of standardization and the scope of the gateway solution.

## Sources of De Facto Compatibility

The origin of de facto compatibility may differ(note: I'm *not* speaking about de facto standardization). See Table 3. The table highlights committee standardization of IT software as a multi-party specification process that leads to a standard. It is a means to coordinate the activities of competing parties.[50] Only if the standard is implemented widely does de facto compatibility result.Compare this with the compatibility achieved by de facto standards. Here, the specification process takes place in a company or in collaboration between several parties. Compatibility results as a by-product of market dominance (e.g., PDF format and Microsoft Windows).

The type of specification process need have no bearing on how ownership of the specification is handled. A company may keep the proprietary technology to itself, monopolise the

[50] Susanne K. Schmidt and Raymund Werle, *Coordinating Technology: Studies in the International Standardization of Telecommunication* (Cambridge, MA: MIT Press, 1998); Martin B. H. Weiss and Marvin Sirbu, “Technological Choice in Voluntary Standards Committees: An Empirical Analysis,” *Economics of Innovation and New Technology* 1 (1990): 111-133.

production of a key component, and define an interface which effectively ties complementary products of other firms to the proprietary technology.[51] Or, a company or group of players may give away its technology with an eye to expected long-term advantages, or enter into coalitions with rivals to enlarge its user base and increase support for its technology. Open source software, for example, usually comes with a non-proprietary, liberal licensing regime.

| **Stages > Type of Specif. Process** | **Specification Process** | | **Market Process** | |
|---|---|---|---|---|
| | **Participation** | **Outcome** | | |
| **Committee Standardization** | Multi-party | Standard | Implemented widely? | Yes > de facto compatibility<br>No > local or no compatib. |
| **Software Development** | Multi-party (e.g., Open Source) | Specification | Market dominance? | |
| | In-company | | | |

Table 3: Two types of specification processes may lead to de facto compatibility between software.[52]

# Compatibility Dimensions

In the following, I discuss the different compatibility strategies. To my knowledge, this has not been done before. For

[51] David and Greenstein, "The Economics of Compatibility Standards."

[52] Source: Tineke M. Egyedi, "Strategies for de facto Compatibility: Standardization, Proprietary and Open Source Approaches to Java," *Knowledge, Technology, and Policy* 14, no. 2 (2001): 113-128.

purpose of reference, I start with dedicated gateways, which is the default strategy in most situations.

## Dedicated Gateways

As defined earlier, a dedicated gateway is a device or convention that allows a limited number of subsystems to be used together. The AC/DC rotary converter, which linked the subnetworks of direct and alternating current in the early years of electricity[53] is an example, as is the Nordunet Plug[54]. This protocol provided access from different subnetworks (i.e., OSI/X. 25, EARN, DECnet, and ARPANET/IP) to a shared backbone. Both these gateways were designed to link specified subsystems.

Different views exist about the degree of flexibility which dedicated gateways provide. Hanseth emphasises the flexibility they create for experimentation at subsystem level and their importance in the phase of system building.[55]

On the other hand, these gateways work as ad hoc solutions, often worsening subsystem entrenchment. Although they may initially offer flexibility, they may turn out to be ". . . another instance of a temporary solution to the consequences of inflexibility. . . . If gateways are . . . [not standardized or modular],

---

[53] Hughes, "The Evolution of Large Technological Systems."

[54] Ole Hanseth, "Gateways—Just as Important as Standards: How the Internet Won the 'Religious War' about Standards in Scandinavia," *Knowledge, Technology, and Policy* 14, no. 3 (2001): 71-90.

[55] Ibid.

. . . they may add the sort of complexity to the infrastructure that obstructs flexibility."[56]

## Standardization

As said, committee compatibility standards are generic gateway solutions. They create complements and facilitate substitution between standardized artefacts. For example, widespread use of the ISO standard for freight container dimensions created a technology—i.e., transport mode—neutral system environment. Moreover, it also created a supplier-neutral system environment (i.e., generic in the economic sense) by means of a level playing field for different vendors. Indeed, since the early days of the computer, customers have been tied to the products of their initial platform provider and have not been able to switch systems without incurring heavy costs. Dedicated interconnections between proprietary systems only partly alleviated the interoperability problem. Although technically feasible, such interconnections were too costly, numerous and cumbersome to create and sustain. In the 1980's this resulted in standards activities which focused on "open systems." Open systems are ". . . computer environments that are based on de facto or international standards, which are publicly available and supplier independent."[57]

---

[56] Duncan, "Capturing Flexibility of Information Technology Infrastructure," 49.

[57] Dinklo, "Open Systemen," 29-30.


Figure 2a: Standardization dimension.

Figure 2a projects degree of standardization on a dimension. Technical compatibility achieved by an ad hoc, improvised solution is portrayed to the left (i.e., no standardization). Dedicated gateways and proprietary de facto standards are categorised as such on this dimension. Highly standardized solutions such as reference models would be projected to the extreme right.

## Modularity

To specify the term modularity, "A system is modular if it consists of distinct (autonomous) components, which are loosely coupled with each other, with a clear relationship between each component and its function(s) and well-defined, . . . [58] interfaces connecting the components, which require low levels of coordination."[59] In ICT modularity plays at different system

[58] Wolters includes "standardized interface" as a property of modularity. However, I agree with the comment of my colleague, Jos Vrancken, that "the presence of an interface is far more important than their being standardized."

[59] Matthijs J. Wolters, "The Business of Modularity and the Modularity of Business" (doctoral dissertation, Erasmus University of Rotterdam, the Netherlands, ERIM Ph.D. Series in Management no. 11, 2002).

levels. Software modules may be used in what Reitwiesner and Volkert[60] call componentware (component-based software) or, at a higher level, in pick-and-mix configurations. Modularity is the second compatibility dimension. See Figure 2d. On the left end of this dimension, the modular approach is not applied. "Improvised" solutions would be projected here (low degree of modularity). On the right end highly modular approaches are projected. The framework or reference model indicates which components or modules are included and how they are interrelated.


Figure 2b: Modularity dimension.

## Interactive Compatibility

The term '(interactive) compatibility artefact' is used here to refer to technical devices and conventions that create compatibility between ICT components and (sub)systems. For example, an interface, middleware, gateways and software agents. Middleware refers to a generic building block that supports

---

[60] Bernd Reitwiesner and Stefan Volkert, "On the Impact of Standardization on the Provision of ERP-Systems as Mission Critical Business Infrastructure," in *Standards Compatibility and Infrastructure Development: Proceedings of the 6th EURAS Workshop,* eds. K. Dittrich and Tineke M. Egyedi (conference held at Delft University of Technology, Delft the Netherlands, June 28-29, 2001) 183-202.

different applications (e.g., DirectX creates 3D images in computers games; web services communicate between applications; the Java platform is used to create a vendor-independent programming environment). Gateways usually create compatibility between protocols in a fixed, static way. However, they sometimes also negotiate compatibility in a more dynamic manner. Krechmer and Baskin[61] use the term adaptability standard to capture negotiation between standardized telecommunication services: "Adaptability standards specify a negotiation process between systems which include two or more compatibility standards or variations and are used to establish communications. These standards negotiate the channel coding and/or source coding. (...) Examples include: T.30 (used with G3 facsimile), V.8, V.8bis (used with telephone modems), G.994.1 (used with DSL transceivers), and discovery protocols."

The potential relevance of negotiating compatibility also applies to non-standardized settings. In the future, agent technology may also play an important compatibility-forging role. Specific attributes of software agents are that they are autonomous, goal-driven and can negotiate and interact with their environment (i.e., can communicate, act and react on their environment.[62] These features are essential to intelligent gateways. Although the technology is still largely in the research phase, one

---

[61] Ken Krechmer and E. Baskin, “Standards, Information and Communications: A Conceptual Basis for a Mathematical Understanding of Technical Standards,” in *Proceedings of the 2nd IEEE Conference on Standardization and Innovation in Information Technology*, SIIT 2001 (conference held at University of Colorado, Boulder, CO, October 3-5, 2001), 106-114.

[62] Marijn Janssen, “Designing Electronic Intermediaries: An Agent-Based Approach for Designing Interorganizational Coordination Mechanisms” (doctoral dissertation, Delft University of Technology, 2001), 11.

could imagine a future in which these agents are designed to self-organize compatibility and manage the complexity of conversion for the sake of interoperability.

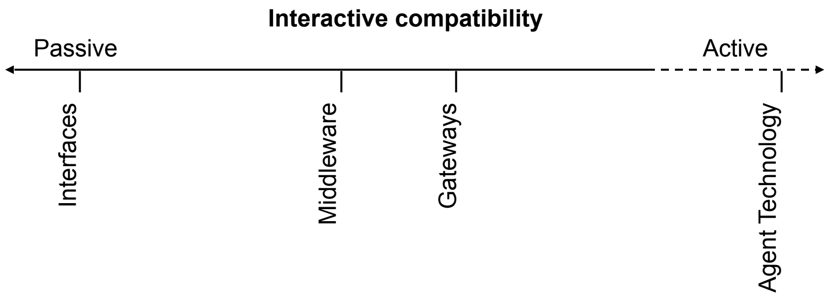


Figure 2c: Interactive compatibility dimension.

In Figure 2c, the compatibility artefacts are mapped onto the dimension of interactive compatibility. This dimension identifies artefacts as more passive or more active in forging compatibility. At the high end of this dimension, artefacts are projected that have the capacity to negotiate and interact in an intelligent and autonomous way (e.g., agent technology). At the low end, artefacts are projected that create compatibility in a passive (i.e., static and fixed) manner.

# Compatibility Space

The three independent compatibility dimensions are depicted in Figure 3. The figure illustrates that each (interactive) compatibility artefact, depicted on the X-axis, can be standardized (depicted on the Y-axis) and designed in a modular way (depicted on the Z-axis). But this need not be so.

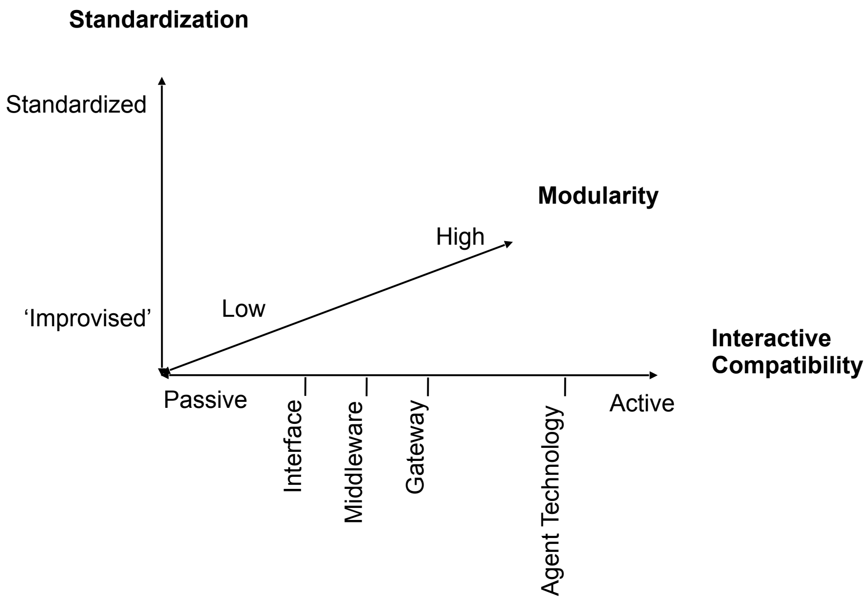


Figure 3: Three-dimensional Space of Compatibility.

The figure further draws attention to the necessity of distinguishing standardization from modularity. And most important, the figure draws attention to as yet unexplored compatibility strategies. At present most artefacts are dedicated, improvised solutions to problems of interoperability (i.e., non-standardized and non-modular). The reader is invited to reflect on a future with standardized agents and modular standards.

Each situation may need a different compatibility solution. By identifying the main dimensions, it becomes easier to discuss and prioritise compatibility solutions.

# Matching Compatibility With Flexibility

Recapitulating, there are several ways to create system flexibility. Experience informs us that specific flexibility objectives are usually better achieved in certain ways than others. For example, committee standards further exchangeability; and modularity facilitates the extension and upgrading of systems.

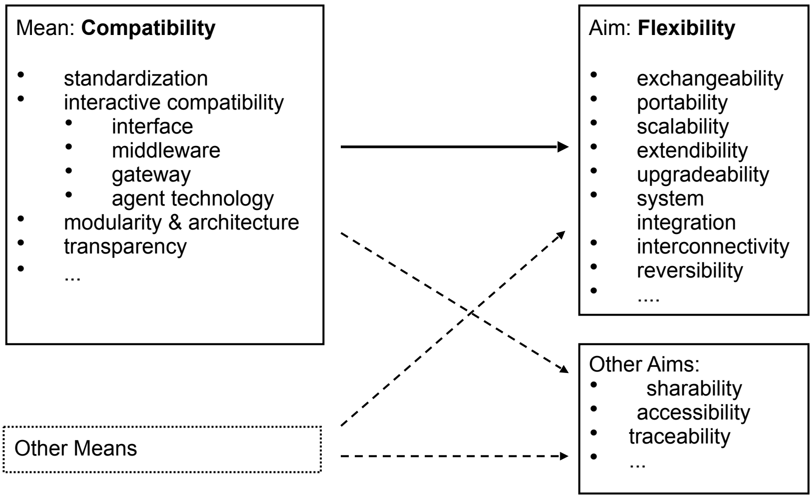


Figure 4: Elements to take into account when seeking a match between compatibility means and flexibility objectives. Although flexibility aims might be achieved by different means, the focus is here on what compatibility strategies may contribute (straight arrow).

Figure 4 models the relationship between compatibility and system flexibility as a causal one. Both categories comprise very different instances. To illustrate the model's relevance, a classic IT problem is that customers are locked-in to a specific computer platform. One solution is platform-independent computing. This would bring about system flexibility (e.g., portability and scalability). In the 1990s, the Java community developed a middleware solution to address this problem, i.e., the Java platform. Would formal standardization of the Java platform, which was attempted twice but failed, have furthered the aims of portability and scalability? Or could a certain agent technology also have solved the problem? Figure 4 may help identify the different options.

In large technical systems, matching compatibility strategies with flexibility objectives may become a complex matter because the choice of strategy will depend partly on factors such as

- whether the system environment is very dynamic (if not, then a dedicated solution rather than a multi-party standard may suffice);
- for what period the solution is foreseen (i.e., how long the overall system is likely be useful); if necessary for the short term, a dedicated solution can suffice; if longer, a more durable solution such as standardisation and modularity may be better;
- at what system level(s) flexibility can be achieved. Can it be achieved at different levels? Does the type of flexibility differ per level, and should the compatibility solution at these levels therefore also differ?

# Conclusion

Compatibility is a core issue in the evolution of large technical systems. Where socio-technical entrenchment appears to hinder a transition, standards and other compatibility strategies are important to look into. They can be a means to (re)create the flexibility required for change.

In this paper, I emphasised that different strategies exist for creating compatibility. These strategies can be plotted as coordinates in a three-dimensional space of compatibility, with the dimensions standardisation, modularity and interactive compatibility.

Matching types of flexibility objectives with compatibility strategies cannot be done in a uniform way. Although some strategies generally seem better set to increase system responsiveness to sustainability demands than others (e.g., highly modularised, modelled consensus standards), the ideal match depends on the circumstances (e.g., what system level is targeted, whether the pressure for change is likely to persist, etc.).

The foregoing illustrates that addressing compatibility problems in an early stage of system design is crucial for the degree of system flexibility later on. In a pre-conditional manner, the chosen compatibility strategy co-determines the ease with which change can take place. Technology innovation policies of governments and companies should therefore take compatibility issues into account, and incorporate a standards—or rather compatibly—policy.

Some very fundamental research questions remain to be answered. Are the three compatibility dimensions proposed the most relevant ones? How generalisable are they and the flexibility objectives to other LTSs?

*Tineke M. Egyedi (PhD, 1996) is senior researcher Standardization at the Delft Institute for Research on Standardization. Her current research interests include how standards affect innovation. She has participated in several EU projects (e.g. Study on the specific*
*policy needs for ICT standardisation) and science foundation projects. She publishes widely. Her most recent book is on 'Inverse Infrastructures' (2012).*
*She has organised and chaired several conferences, and is regularly invited internationally as a speaker. In 2009 she initiated 'Setting Standards', a simulation exercise currently used by NIST to train US policy makers and earlier by NEN for Chinese standardizers. She is vice-president of the European Academy for Standardization, Fellow of the Open Forum Academy, expert to UN/ECE WP 6, and member of ISICT/ IEEE ComSoc's Emerging Technologies Committee.*