



**Who needs
~~use cases~~
~~scenarios~~
applications
*anyway?***

“Asking users what they want is a clear road to [failure].”

Jeff Burke
UCLA School of Theater, Film and Television
NSF Center for Embedded Networked Sensing (CENS)

NSF Future Internet Summit – Arlington, VA – 10/13/2009

Probably true.

But what do we mean by *users* anyway?

Let's focus on systems designers (taken broadly) who make detailed choices about networking as part of their designs...~~and screw them up.~~

Important way to learn from creators is to understand their *process*, not just their products.

What would it be like to study and *facilitate design processes*, as an alternative to use cases?

Important: These processes reflect past, current, and future applications.

Design-ability: A superset of manage-ability?
Or slicing several of the attributes differently?

Observation #1

Who are the 'authors' on (of) the network?

- architects (like this group),
- providers who implement (and fuss with) standards,
- those that design and deploy networks as parts of larger 'systems'.

But more than 'application developers', they're building networks (physical and virtual) that are deeply intertwined with other systems and challenges in a way that I suspect was less true in the past.

This group seems to be struggling mightily, across a variety of disciplines, to keep track of the mapping between what's important in their design process and network elements.

Observation #2

What can be learned from the third class of users *are not only the explicit network requirements of their applications,* but **the design processes of which the network is now both component and facilitator.**

Network design and deployment is increasingly tightly coupled with other design processes, which are motivated by project-specific principles, constraints, and ontologies.

This is especially true in cyber-physical systems, but I suspect applies in other domains as well.

Possibility: *Talk with application people about how they make things, not just what they make.*

Observation #3

Unlike other infrastructure, networks have potential to embed knowledge *that may only be gathered in one 'place' at the time of design* that impacts everything from physical topology, to addressing, NAT and many other things.

Could architectural components help network owners, designers, and users discover and manage their own specific knowledge about specific networks, with no loss in generality?

Possibility: As our architecture(s) improve, could they better embody and communicate design decisions and principles that are increasingly hidden by complexity?

Loose take on overheard examples

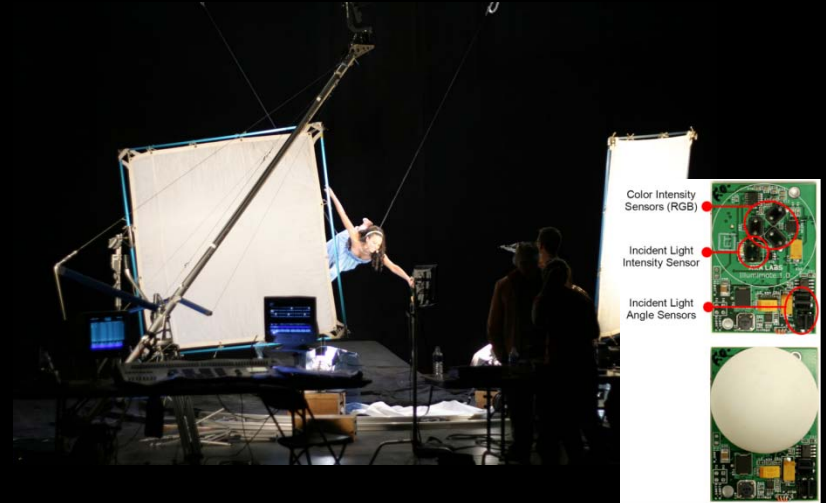
Could supporting energy, economics, or location in, say, routing be treated as specific cases of a more general approach to incorporating other design spaces into network processing?

One person's context is another's content is another's payload: Does that mean the network should support the lowest common denominator, or multiple perspectives (not layers) on the same packet?

What I do (somewhat)



Sensing systems in live performance.



Advanced media production.



Community-focused media.



Interactive media in the built environment.

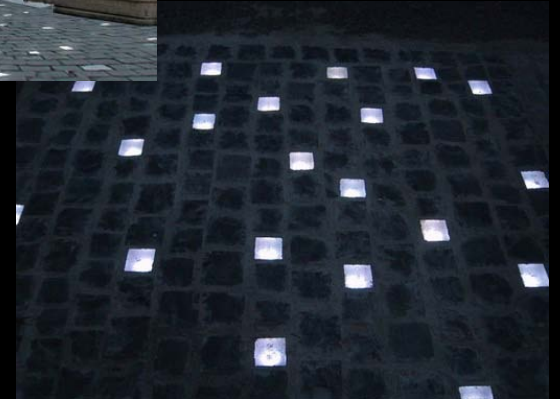


National Center for The Preservation of Democracy
Japanese American National Museum, Los Angeles, CA

Highly designed environments

Space is and will be shaped by networks of media (and sensors, and information) as well as physical construction.

There will be more and more instances of co-design, with more complex physical, social and economic coupling.



Highly designed environments (intersecting with mobility!)



Genting City of Entertainment, Malaysia

How to (re)design and (continuously) operate these hybrid environments?

‘infrastructure’

The (current and future) internet infrastructure is fundamentally different than infrastructure any we have seen before.

It doesn't have a particular material nature, only the abstractions that this community creates.

It organizes access to information as well as to physically distributed devices according to those abstractions.

Would it make sense for developers to be better able to provision and manage additional abstractions?

‘infrastructure’

Other materials and infrastructures *cannot* by their nature intrinsically incorporate information about how their subcomponents were designed and connected.

They can’t remember their mapping between principles (or business logic, or policy, if you like) and their physical reality.

They can’t adopt aspects of the ontologies of their creators, while retaining generality at an underlying level.

The internet *can... does somewhat... could...*

Why should it?

‘infrastructure’

Design processes are related to TCO, time, and innovation.

For example, in theme parks, nearly 50-70% of construction, cost is committed in blue sky, concept design, schematic design. *Big decisions are made with lots of supporting data and guiding principles that affect systems.*

Can we facilitate the persistence and evolution of design knowledge throughout the life of a network?

Can we enable the use of application-specific design abstractions for addressing, naming, routing, etc. within a network that participates in the internet?

(Or do we just push this upwards and make application developers do it over and over again, usually badly and incompatibly?)

‘infrastructure’

“[S]hared visions of the possible
and acceptable dreams of the innovative.”

(L.L. Bucciarelli in Star, 1999).