The Web of Things:
Extending the Web into the Real World

Dave Raggett, W3C

SOFSEM 2010

Contact: dsr@w3.org

Contents

- The Web of Things
- Challenges
- Connecting things into the Web
- Resource Coordination
- The Web of Trust and role of delegation
- New directions for application authoring
- Summing up

Before the Web

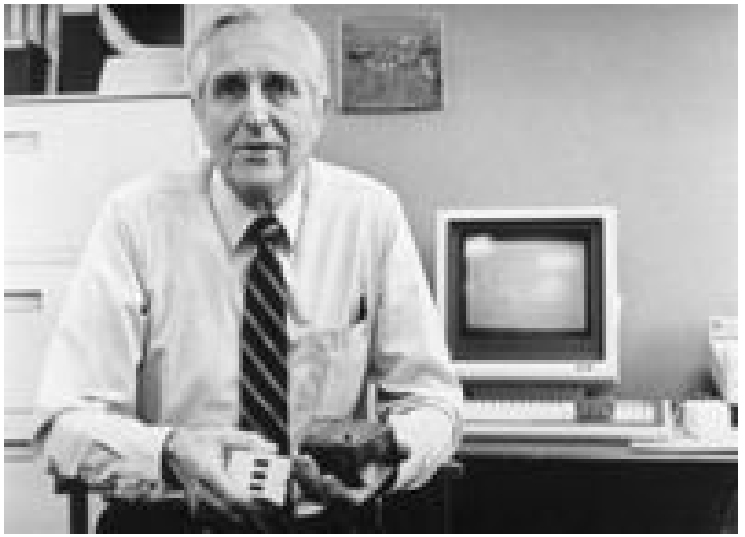
Vannevar Bush

- Scientific advisor to President Roosevelt
- “As We May Think” published July 1945 in The Atlantic Monthly

  A conceptual machine (the Memex) that can store vast quantities of interlinked information

- Same article describes the Cyclops Camera:

  "worn on forehead, it would photograph anything you see and want to record”



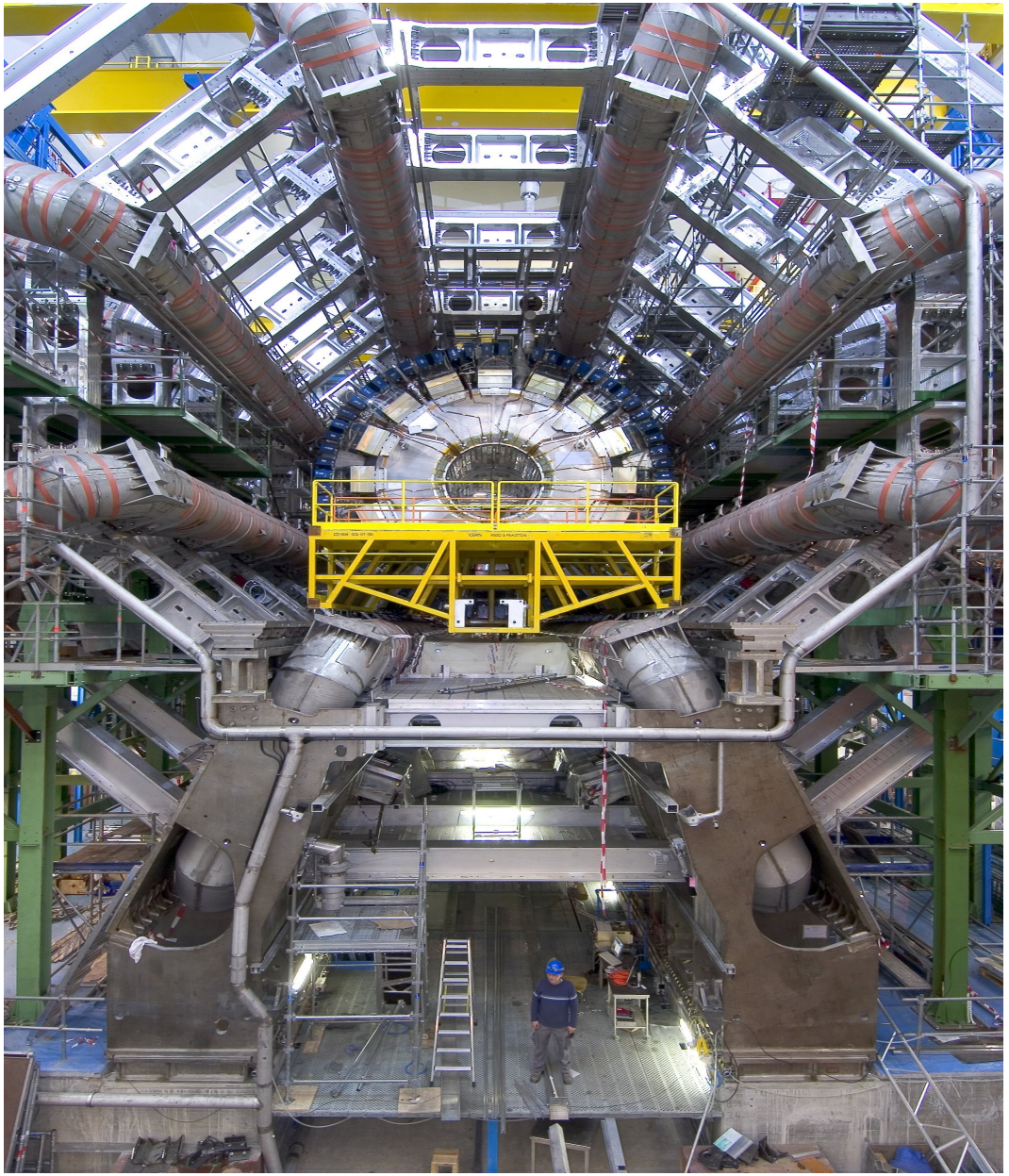
Douglas Engelbart

- mid-1960's Inventor of the computer mouse, he led work on hypertext and graphical user interfaces at SRI International

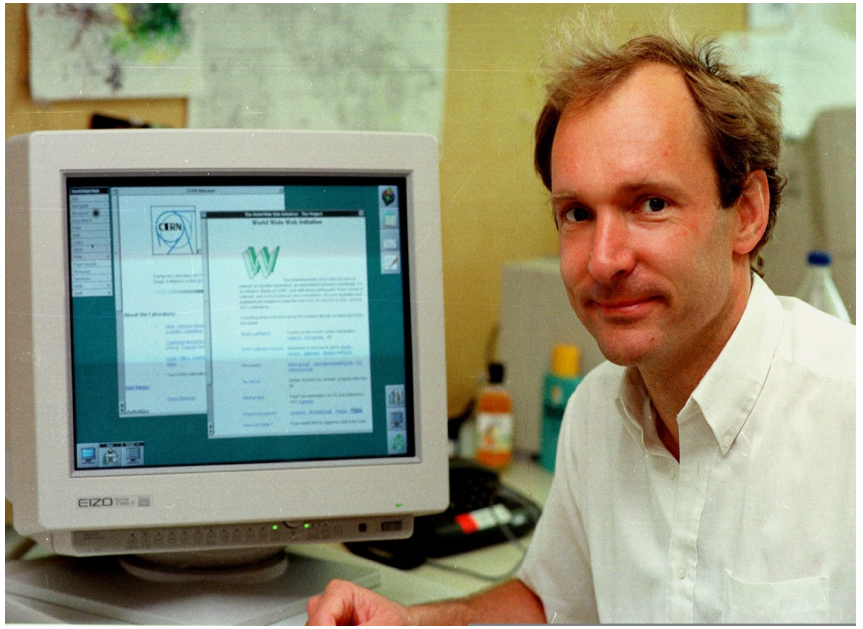
Ted Nelson

- 1965 coins the term “Hypertext”
  - in "A File Structure for the Complex, the Changing, and the Indeterminate". 20th National Conference, New York, Association for Computing Machinery
- Project Xanadu founded in 1960
  - Goal: a networked pay-per-document hypertext database encompassing all written information

CERN – birthplace of the Web



- International research centre for high energy physics located near Geneva
- Large Hadron Collider (LHC) Atlas detector
- Probing conditions at earliest moments of the Universe



Tim Berners-Lee

Inventor of the World Wide Web

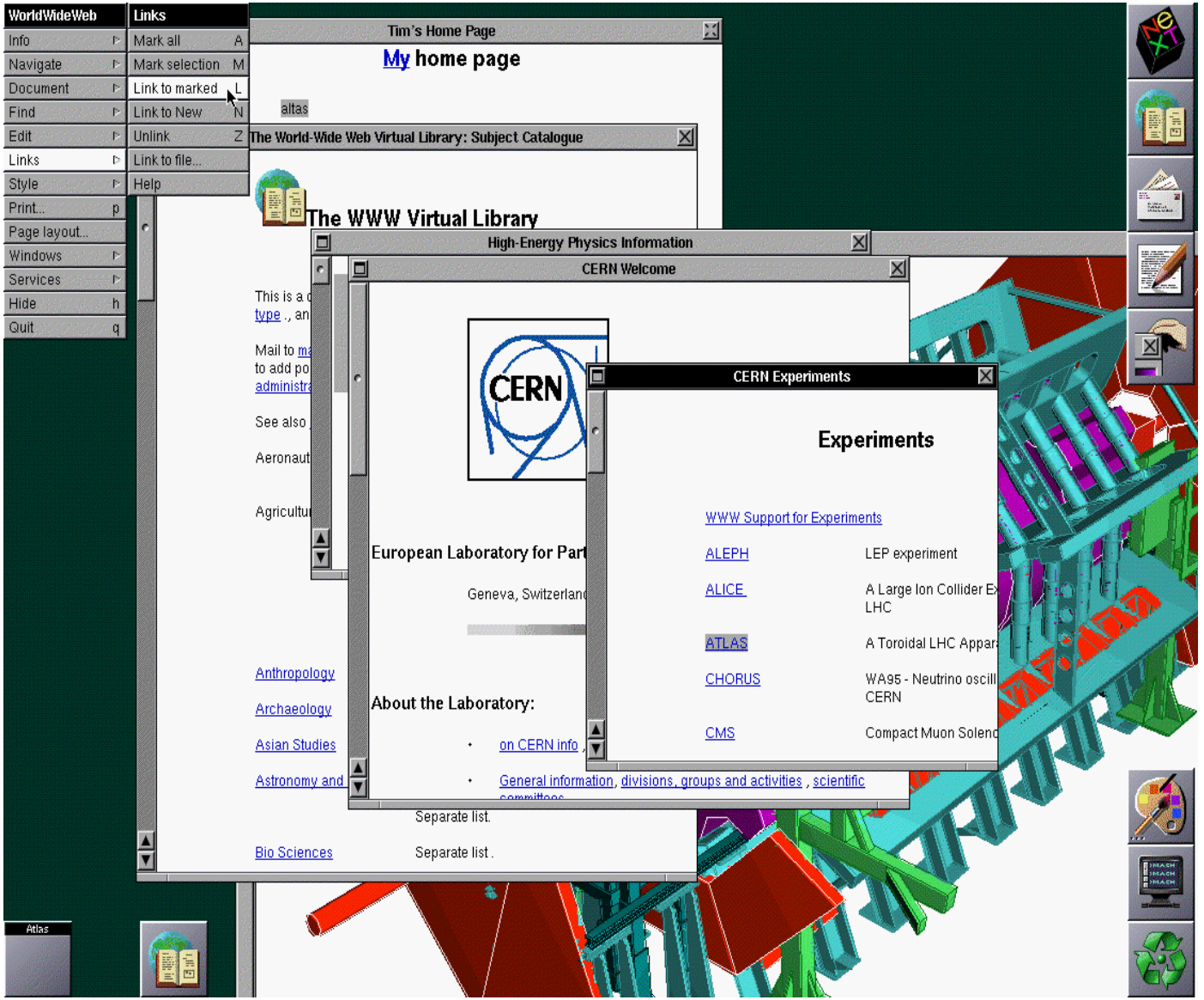
- Friend of a friend at Oxford, we first meet in '92
- 1980 Develops “Enquire” as a simple hypertext system whilst consulting for CERN
- 1989 Project proposal for World Wide Web
- 1994 Founds W3C to lead the Web to its full potential

Enquire

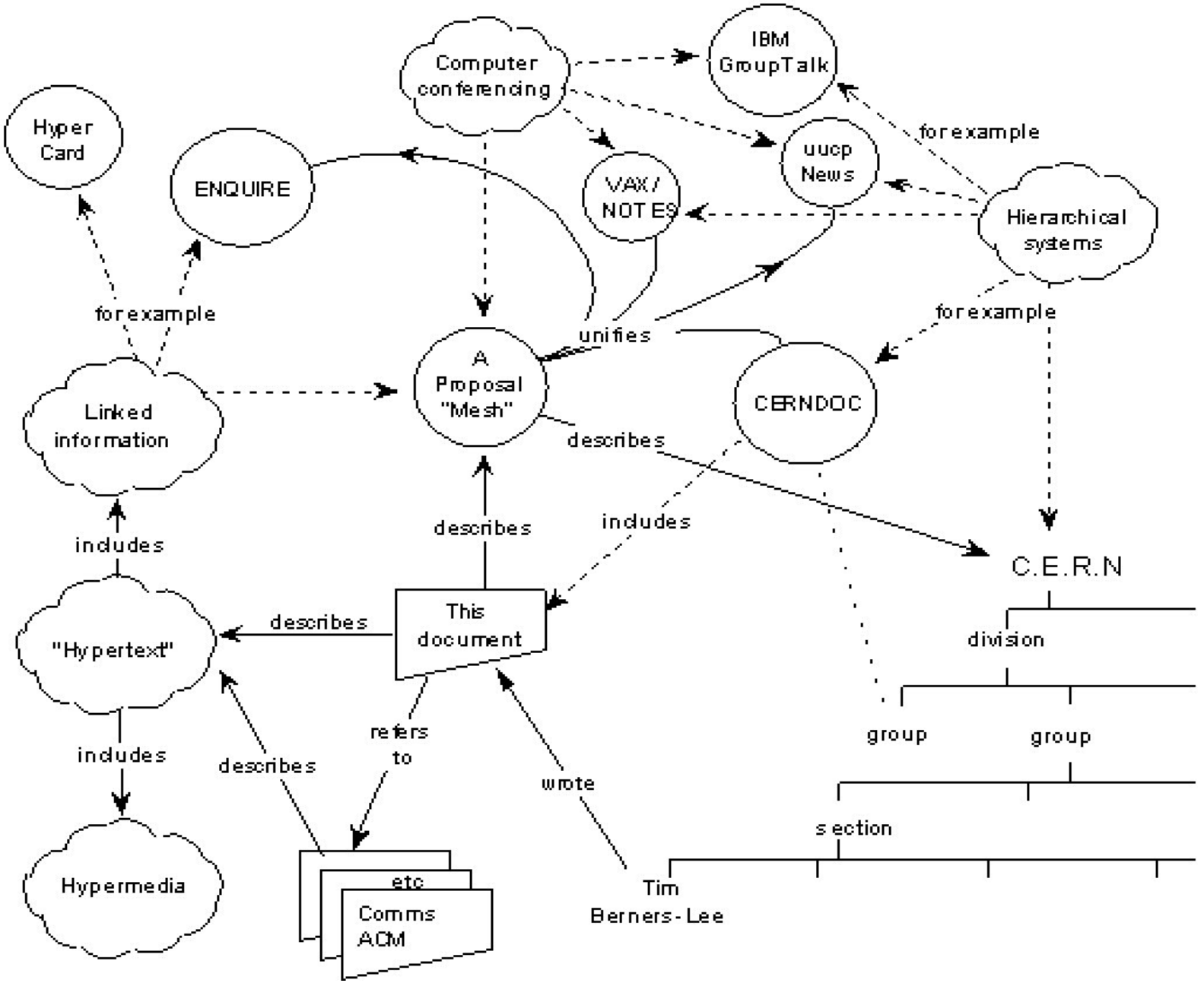
```
> ENQUIRE
Enquire V 1.1

Hello!
Opening file (PSK-PCP)VAC-V1:ENQR...

PSB Vacuum Control System                        (concept)  <   O>
--- ------ ------- ------

 [ 1] described-by: Enquiry System
      An experimental system for which this is a test.

 [ 2] includes: Vacuum History System
      Records and displays slow changes in pressure.

 [ 3] includes: Vacuum equipment modules
      Perform all the hardware interface

 [ 4] includes: Control and status applications programs
      Provide operator interaction from the consoles.

 [ 5] described-by: Controle du System a Vide du Booster 11-2-80
      Operational specification of the software

 [ 6] includes: PSB Pump Surveillance System          PCP 228
      Allows rapid monitoring of pressure changes

[number      ]
```

Early Web Browser

Browser/editor on NextStep workstation



1990: WWW Architecture



Only worth a Poster at Hypertext '91

Hypertext'91 Conference decides that the WWW is only worth a Poster!

Initial Simplicity

- Tim deliberately chose to keep the initial version of the Web really simple to encourage widespread adoption
- Simple hypertext markup (html) with link types
  - Chapter 1
  - Simple protocol (http) with global addresses
- Designed to be rendered on wide range of devices
- Images and other media shown in external viewers

Followed by Rapid Evolution

- Exponential growth in Web traffic
- Addition of capabilities to HTML and HTTP
- NCSA Mosaic as first widely used browser
- Netscape as first Internet boom company
- Microsoft turns on a dime
- Browser wars won by Internet Explorer
- Competition: Firefox, Opera and Safari
- Mobile browsers and XML standards
- Competition with proprietary formats

But what is the Web?

- According to W3C, the Web is
  - An information space in which the items of interest, referred to as resources, are identified by global identifiers called Uniform Resource Identifiers
    - See http://www.w3.org/TR/webarch/

What is the Web?

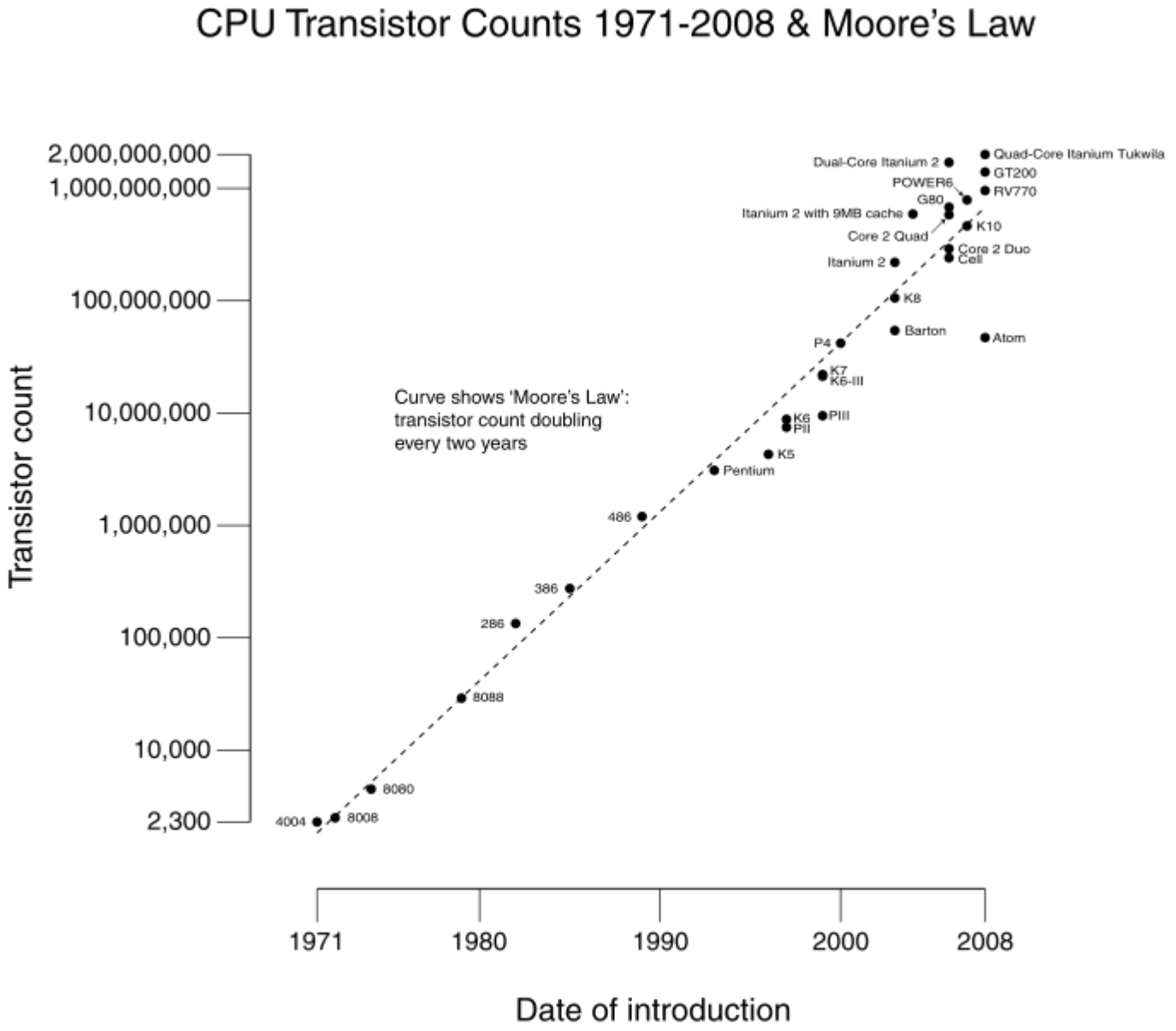
- Earlier version of webarch defined the Web in terms of a system rather than a space
  - Networked information system consisting of agents (programs acting on behalf of a person, entity, or process) that exchange information
    - http://www.w3.org/TR/2003/WD-webarch-20030627/
- But many people just conceive of the Web as
  - The set of HTML pages you can access from a Web browser

Tunnel Vision
and coming out into the sunlight

- Conceiving the Web only in terms of today's browsers is very limiting
  - Undue focus on HTML and browser APIs
- What about
  - Other modes of interaction (aural, tactile)
  - Explosion of new kinds of networked devices
  - Distributed applications & end-to-end models
  - Agents acting on behalf of people
  - Semantic Web of symbolic and statistical knowledge
  - Web of trust and human relationships

The Web of Things

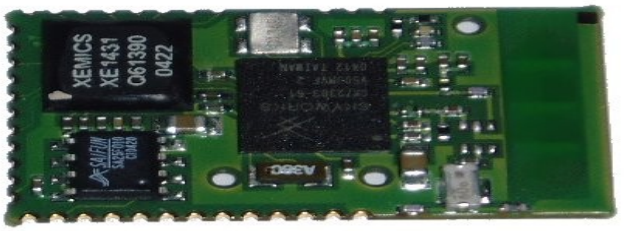
Moore's Law

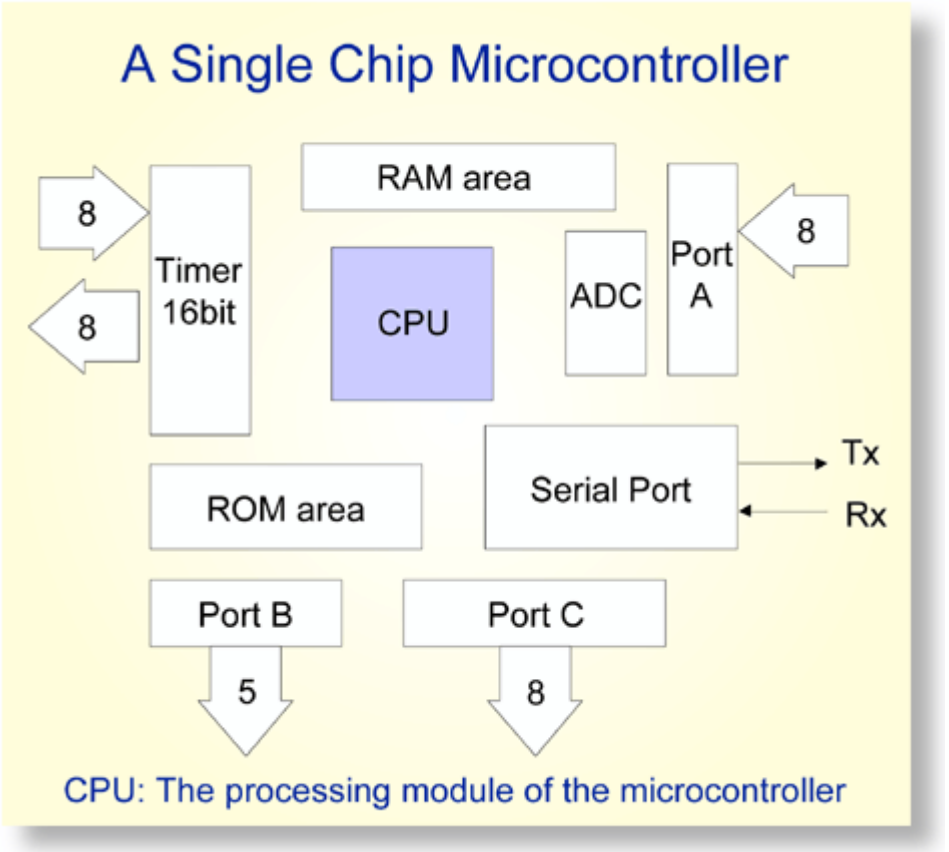

CPU Transistor Counts 1971-2008 & Moore's Law
Transistor count
2,000,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
2,300
Curve shows 'Moore's Law': transistor count doubling every two years
Dual-Core Itanium 2
Quad-Core Itanium Tukwila
GT200
POWER6
RV770
G80
Itanium 2 with 9MB cache
K10
Core 2 Quad
Core 2 Duo
Cell
Itanium 2
K8
Barton
Atom
P4
K7
K6-III
K6
PII
PIII
K5
Pentium
486
386
286
8088
8080
8008
4004
1971
1980
1990
2000
2008
Date of introduction

Interconnect Costs

- Year on year improvements with Integrated circuit technologies and processes
  - Bigger gate counts, or ...
  - Reduced cost for a given gate count
- Moore's law has made it easy to integrate RF circuitry alongside digital circuitry
- Rapidly dwindling incremental cost of networking all kinds of devices

Microcontrollers





- Computer on a chip
- Fastest growing segment of computer industry
- Average home now contains around 200
- Cars between 35 and 100 for luxury models

Uses of Microcontrollers

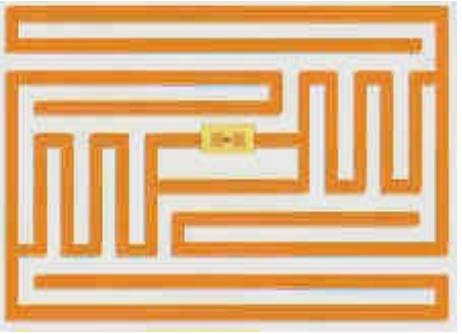
- TV sets, TV remote controls, Video recorders printers, cameras, scanners, fax machines
- Ovens, toasters, refrigerators, washing machines, central heating systems
- Mobile phones, PDAs, MP3 players, computer monitors
- Car body electronics, air conditioning, seat control, chassis and safety, infotainment, power train
- The list goes on and on ...

URIs for physical objects

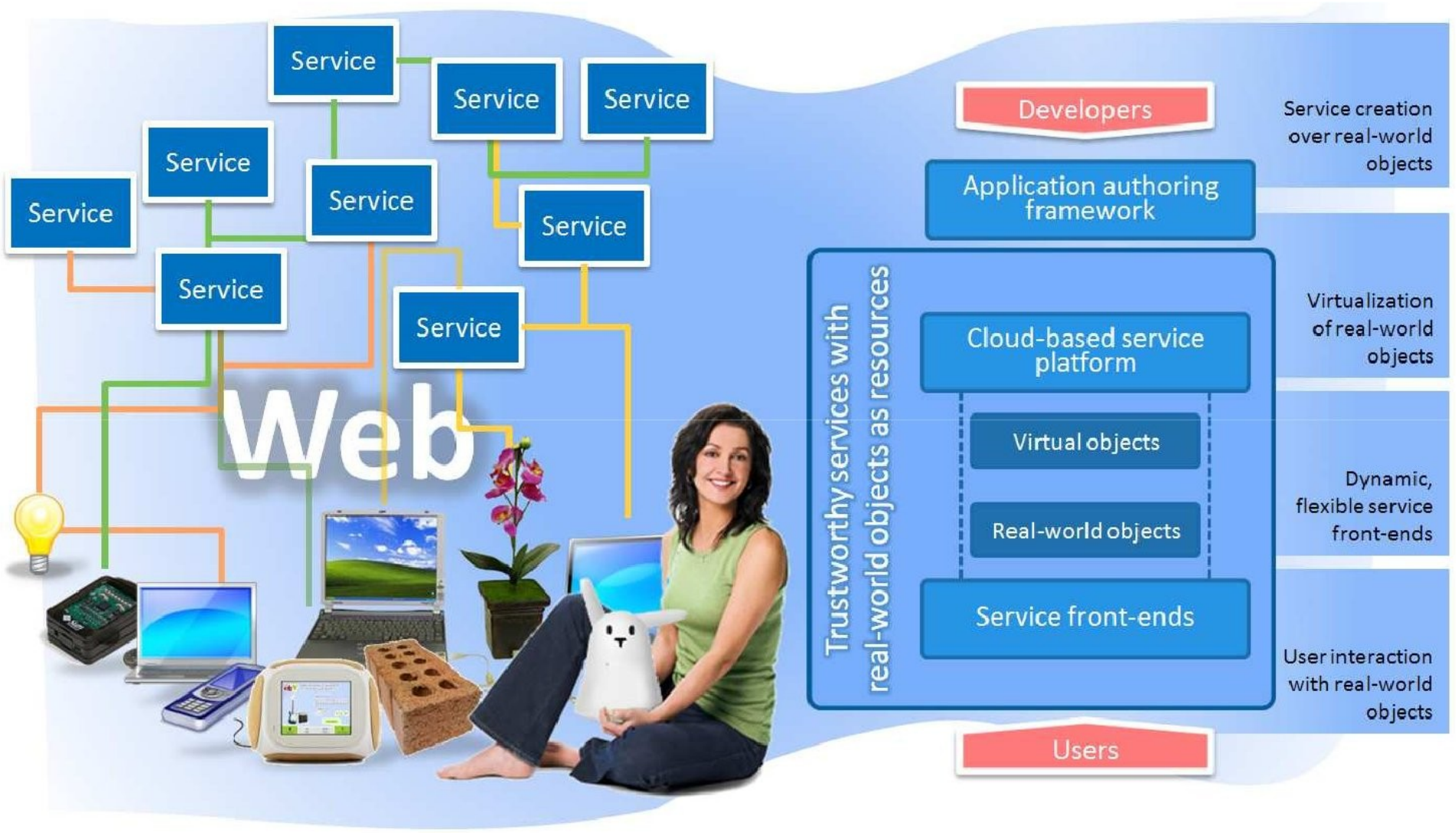
Barcodes as a way to connect physical objects to the Web

RFID

Electronic versions of barcodes
but with extended capabilities



Web of Things

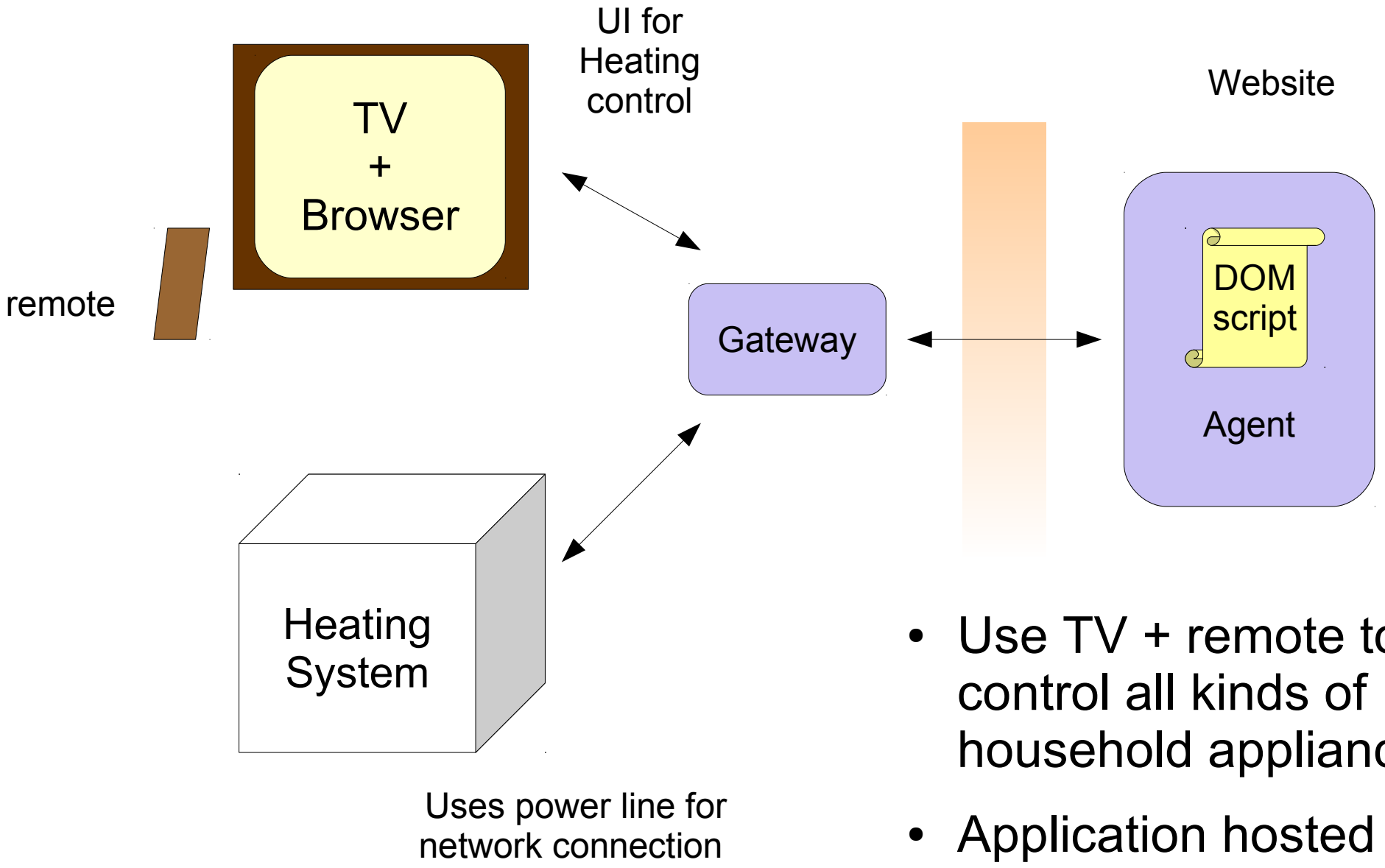


From IOWA Proposal Part B

What's the Value?

- Reduced costs of heating, cooling and lighting for homes and offices
- Improved physical security and peace of mind
- Preventative maintenance in advance of appliances breaking down
- Improved standard of care for the elderly
- Better choices for home entertainment systems
- Fulfilling the potential for applications that combine local and remote services

Home network example



- Use TV + remote to control all kinds of household appliance
- Application hosted by website

Challenges

Programming Barriers

- Steep learning curve for C++, Java, etc.
- Web technologies have proven much easier
  - HTML, CSS, JavaScript, PHP, Python, …
  - Many more developers
- But there are still problems
  - Heterogeneity of devices and software
  - Support for assistive technology

Networking Technologies

- Applications will need to work over a mix of rapidly evolving networking technologies
  - Ethernet over twisted pair
  - DSL over copper phone lines
  - Ethernet over building power wiring
  - WiFi and WiMax
  - Bluetooth
  - ZigBee sensor networks
  - GSM and cellular packet radio
- Further challenge of different addressing schemes, e.g. peer to peer networks

Moving up the Network Stack

- Current academic and industry focus is on networking and low level services
  - Internet of things, not yet the Web of things
- Next stage will be to focus on how to make it easier to create distributed applications
- How to create applications that can work across
  - Different networking technologies
  - Different generations of devices
  - Different vendors
  - Different trust boundaries

Yesterday and Tomorrow

- Devices are replaced on a variety of time scales
  - Mobiles
  - Televisions
  - Heating systems
  - Building infrastructure
- How to ensure that yesterday's services will work with tomorrow's devices and vice versa?
  - Mix of product generations and technologies
  - Need for layered architecture that cleanly separates out different concerns
  - Critical importance of standards

Realizing the Potential

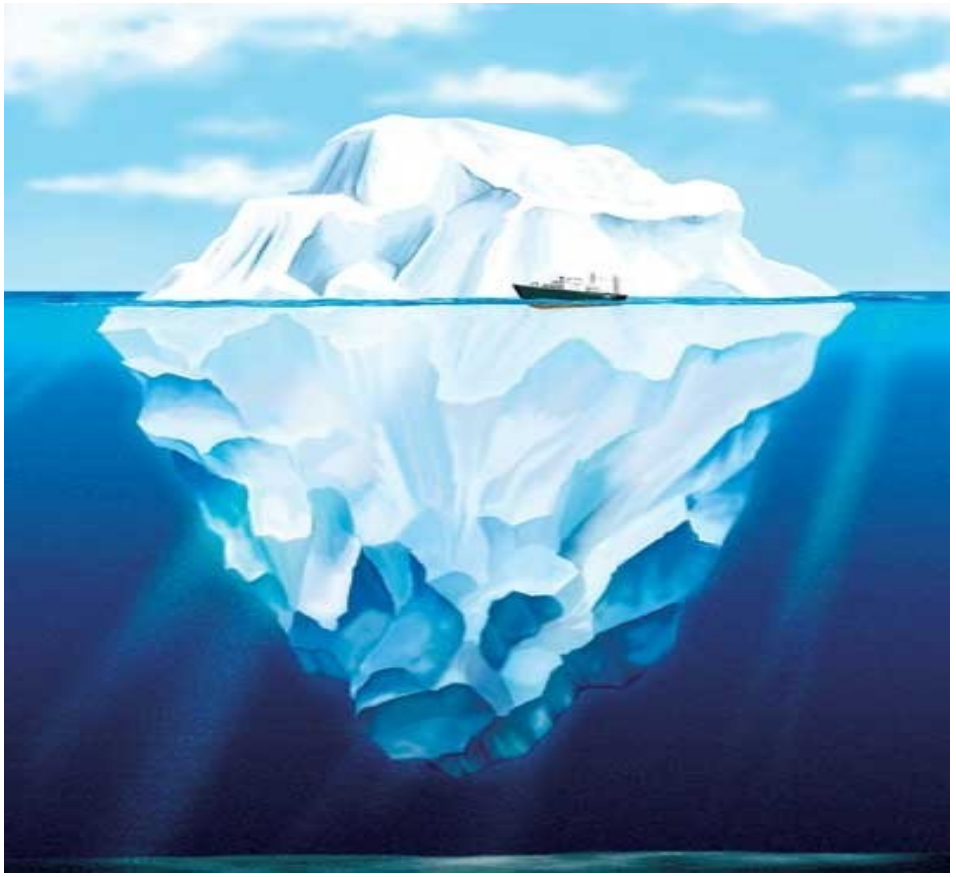
- Initially, just proprietary solutions
  - End user purchases complete solution
  - Single vendor and single product generation
- Followed by narrowly focused industry standards
  - Pictbridge for printing direct from camera when printer and camera from different vendors
  - DLNA for connecting multimedia devices in the home
- Broader standards follow later, enabling new applications
  - Traditional programming languages like C++ and Java offer low level control but are costly to develop with
  - Web technologies will make applications easier and cheaper to develop, enabling a much bigger ecosystem

Connecting things into the Web

Web of Things

- Apply Web technologies to make it easier to create applications of networked devices and services
- Start with a Web abstraction layer that hides the details web authors don't need to deal with
- Connect things into the Cloud for easier authoring
  - Access to device capabilities
  - Respond to input from sensors
  - Drive actuators e.g. to turn up the heating

Simplicity has to be worked at



- Web oriented models of world as basis for easier development of service front-ends

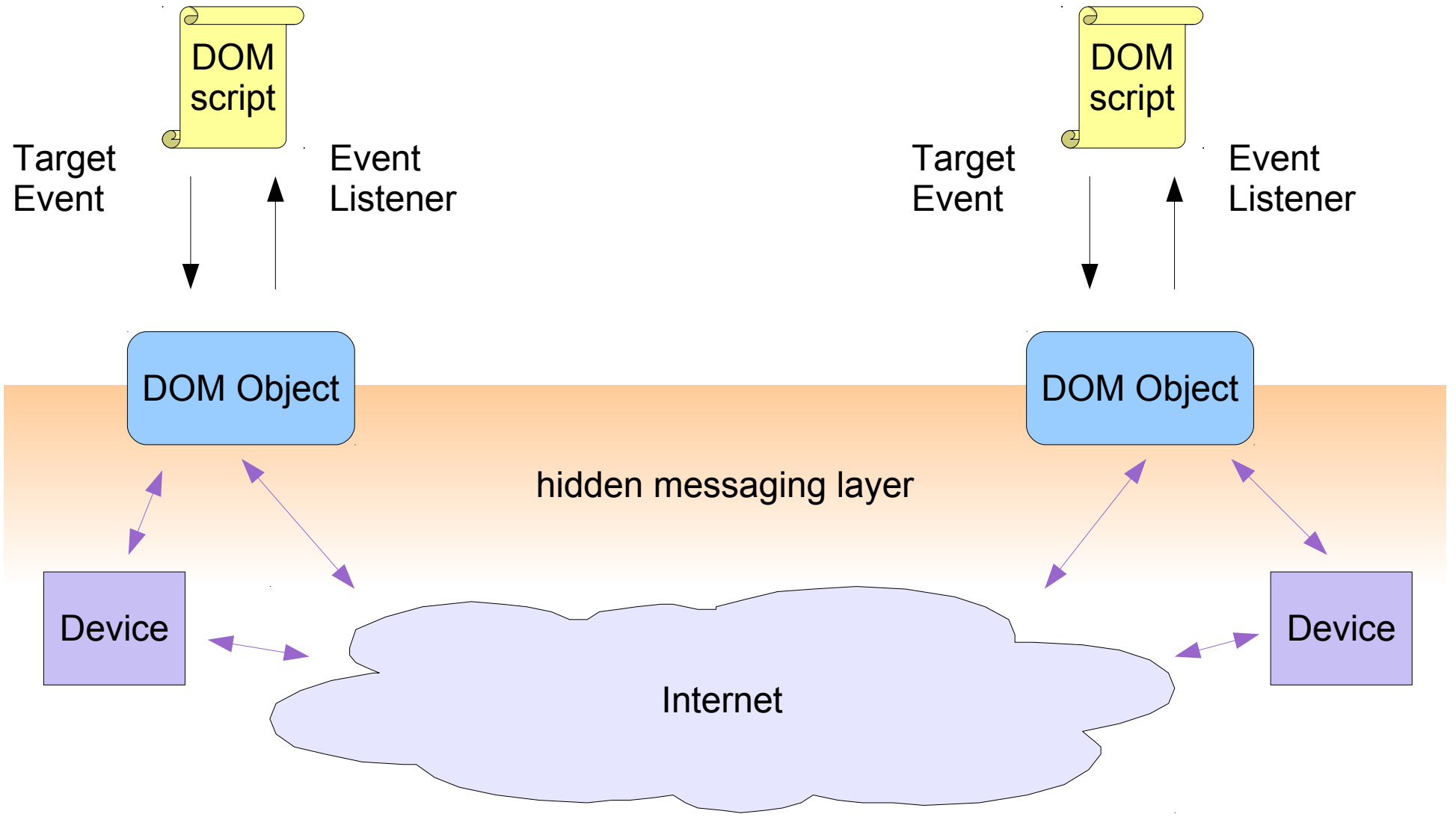
- Hidden infrastructure and associated models

  The unseen part that keeps the rest afloat

Context Awareness

- Models of users, things and their environment
  - What devices are present in my home?
  - What are my personal preferences?
- Virtual objects as proxies for things
  - Hiding the underlying complexity
    - Addressing, routing, synchronising
- Ontologies, APIs and live objects
- Basis for adaptation at authoring and run-time

Proxies for services



Devices

- Many kinds of devices
  - Smart sensors/actuators accessed by HTTP
  - Devices which require readers
    - RFID and barcodes
  - Gateways between different networking technologies
  - Devices with programmable behaviour
    - Via XML, scripts or byte codes
- Device appears as proxy object in web run-time
  - Sensor input mapped to DOM event in web app
  - Target event at proxy object to drive actuator

Web Run-Times

- Execution environment for web apps
  - Mark-up, style sheets, scripts, etc.
- Browsers, Widgets, and now Websites
- Personalized apps that run 24x7 on your behalf
  - Run somewhere in the **Cloud***
  - Combo of mark-up, style sheets scripts, etc.
  - Appear as shareable widgets on web pages
  - Interact with user through real-world things
- *Moving the Web out of the browser!*

* Cloud: dynamically provisioned virtual computing resources

Resource Coordination

Resource Coordination

- How to ensure that devices and services function as part of a distributed application
  - Support for discovery and adaptation
  - Descriptions of devices and services (resources)
    - Including basis for access control, identity and trust
    - Coordination and control
      - Services provided by an individual device or a collaboration between multiple devices
      - Scheduling and fair access to scarce resources
      - Services as an orchestrated sequence of events
      - Error handling and recovery

Resource Coordination

- How does a device obtain an IP address?
  - Zeroconf (UPnP, Bonjour, Avahi, mDNS, SSDP)
- What kind of message routing topology?
- How does a device or service advertise itself?
- How do applications discover and bind to devices and services?
- Cloud of things - provisioning
- How are privacy and security addressed?

Web-based Coordination

- Describing UI and behaviour
  - Markup and/or scripting
- Using URIs to name devices and services
  - Rich meta-data for describing device capabilities and security policies
- Expose device/service as object in local object model
  - Hides addressing/communication details
  - Enable application to continue to work when devices and network technology/topology change

Web-based Resource Binding

- Either name resource or provide description
  - URI for resource name or description, or
  - Explicit description
    - XML element, or
    - meta data (RDF) or
    - scripting API
- Implicit or explicit resource binding service
  - Broker and access control may be local or remote
  - Event when resource is bound and unbound or on access control error
  - On success, resource exposed as DOM object

Policies

- Web app requests access to a device/service
- Broker invokes policy engine
- Policies as rules that express user and provider preferences
  - Over credentials for identity/properties
- Policies may delegate to trusted 3rd party
- Privacy policies define obligations for handling of personal data
  - What can it be used for?
  - How long can it be retained for?

Compositions

- Logical device or service that is a composition of others
  - Means to configure devices and services to work together
    - Copier = camera + printer
    - Commands = microphone + speech recognizer + command grammar
    - Gestures = camera + video processor
- Composition treated as logical device with its own description in the context models
- Compositions can be nested as needed

W3C Delivery Context Ontology

- Ontology covering user preferences, device capabilities and environmental conditions
  - Modular design for scalability
  - Exposed through client and server-side APIs
    - These are being worked on in parallel
  - Coordinated effort to avoid inconsistent models
    - Success story for device orientation
    - Too late for conflicting treatment of pixels
  - Current focus on mobile devices
    - Other kinds of consumer devices expected next
    - New work started on personalization (accessibility)

Control and event routing

- Strongly coordinated, no delegation
  - Controller manages access to services
  - All events routed through control point
- Weakly coordinated, partial delegation
  - Controller manages access control, but delegates event management
- Uncoordinated, full delegation
  - Peer to peer communication model
  - Devices responsible for resource management

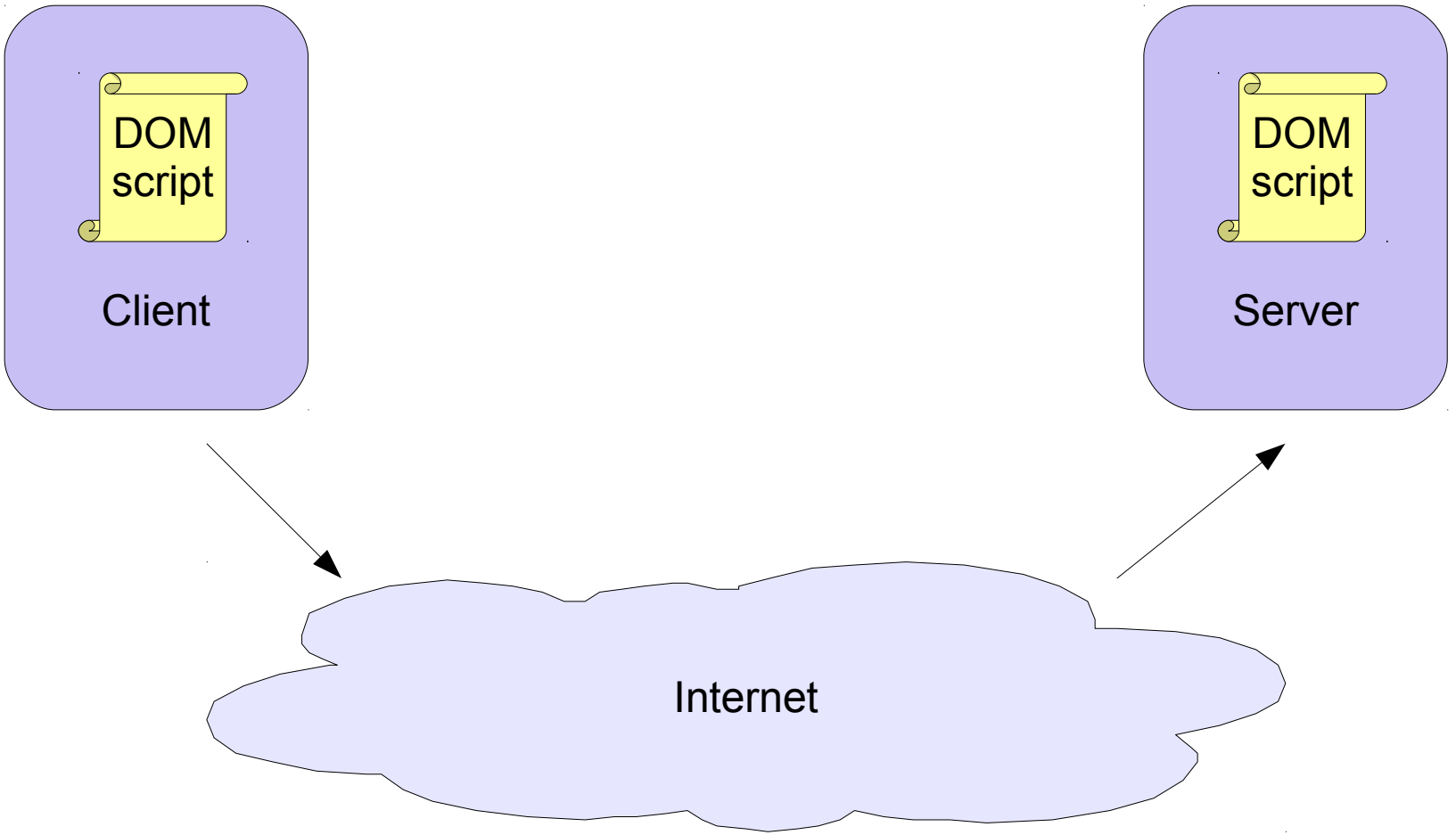
Agents not Web Pages

Event Transport

How to deliver events to devices?

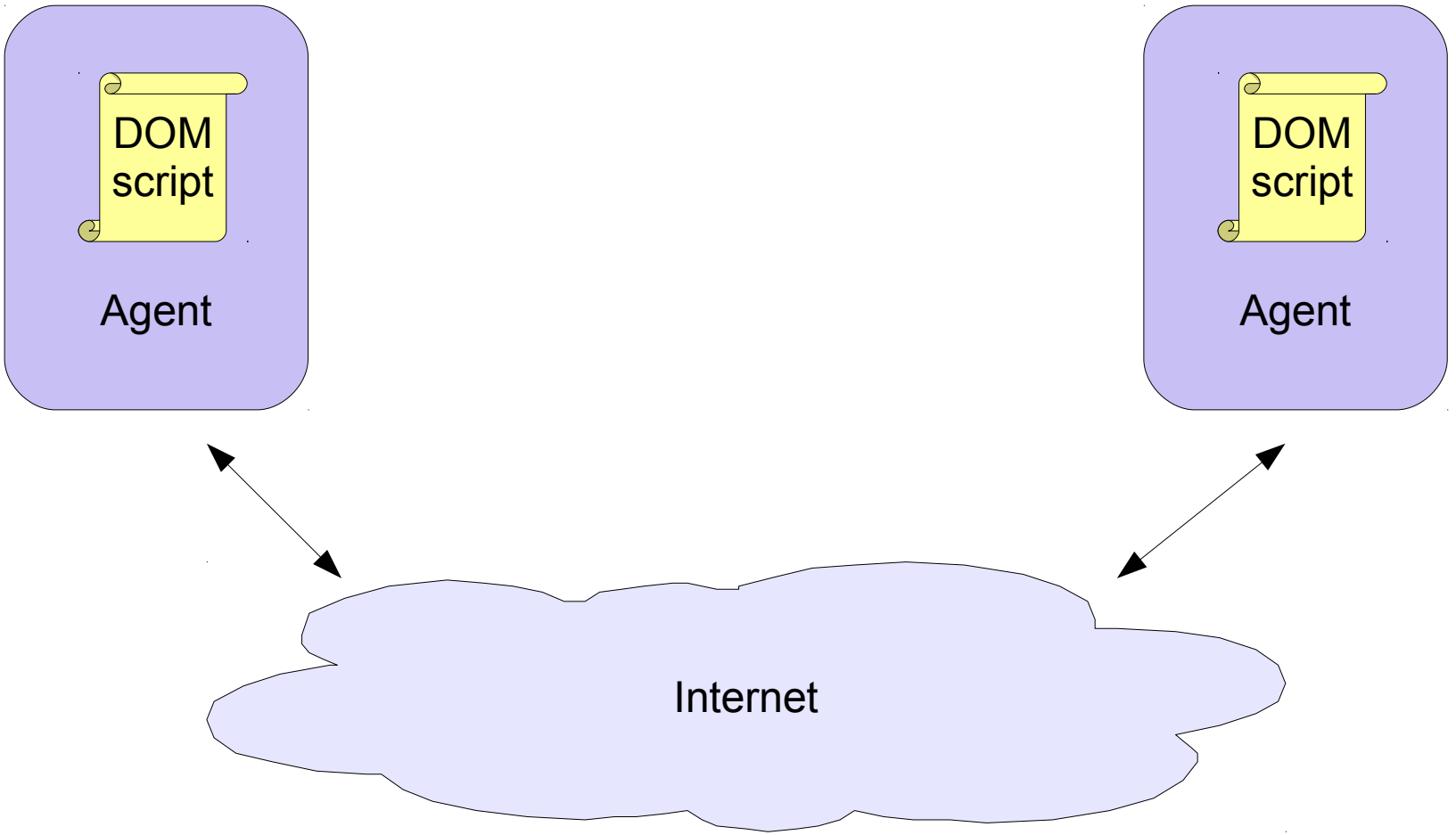
- Firewalls are intended to block undesired traffic
  - No incoming HTTP connections by default
- Evolution of mechanisms to tunnel events through Network Address Translation
  - STUN, STUNT, TURN, etc.
  - Skype and success at a cost
- Bindings to event transport protocols
  - HTTP, SIP, XMPP

Client or Server?



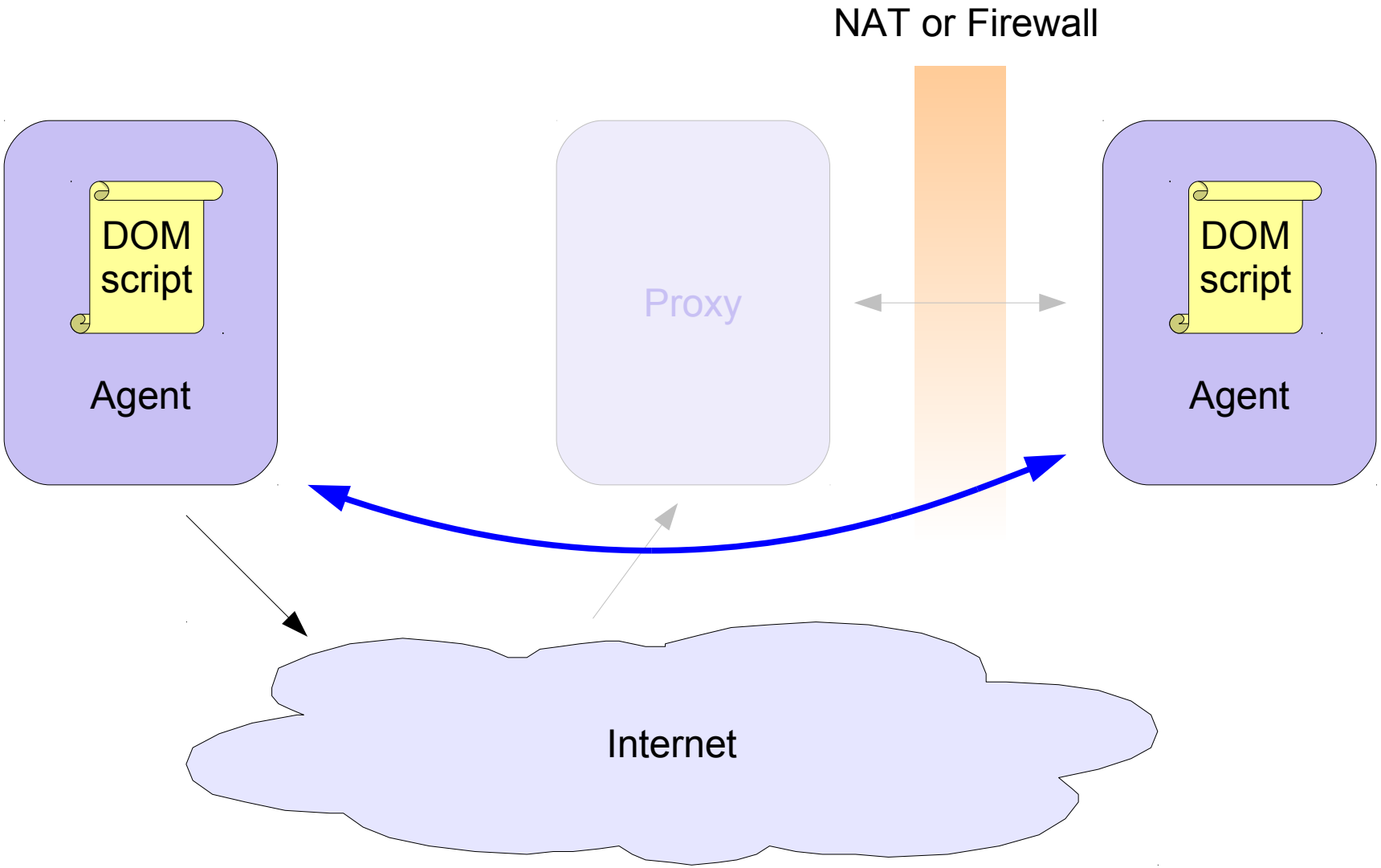
Client or Server?

Agent combines client and server



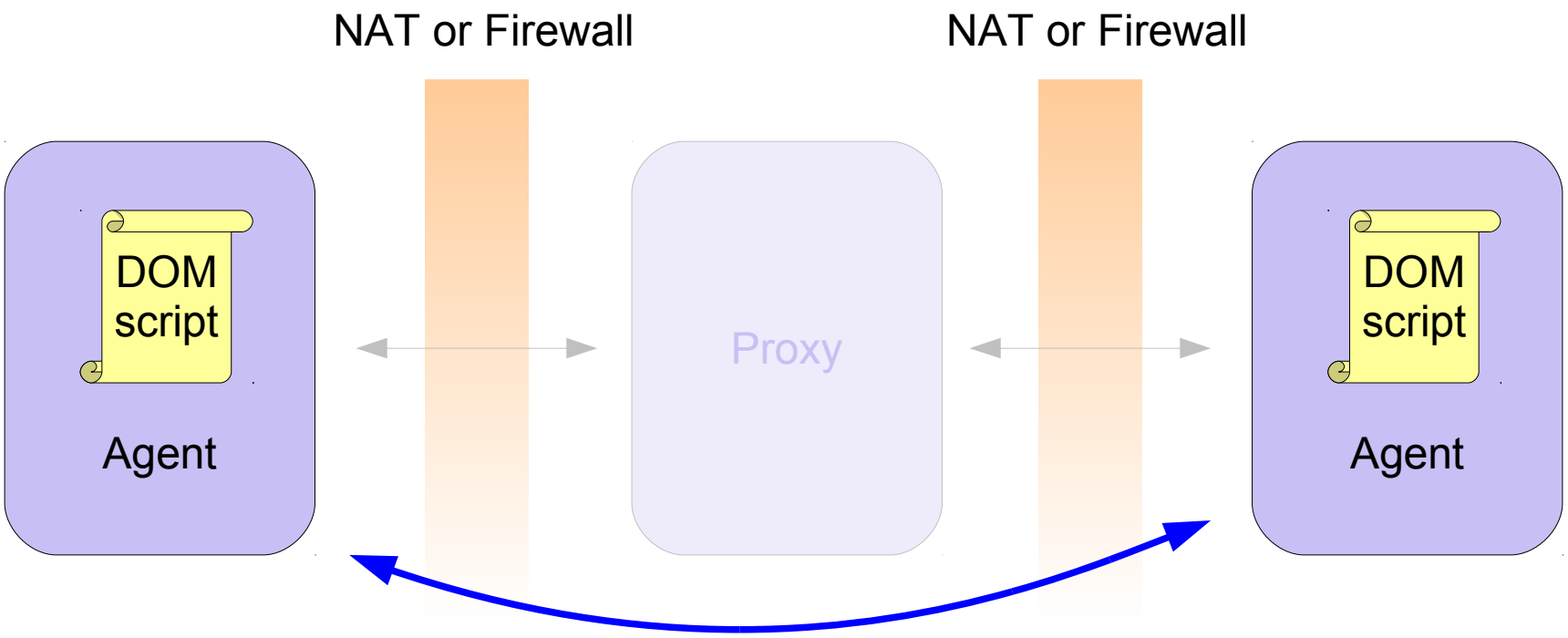
Tunnelling through NAT

Proxy may arrange for direct link through NAT



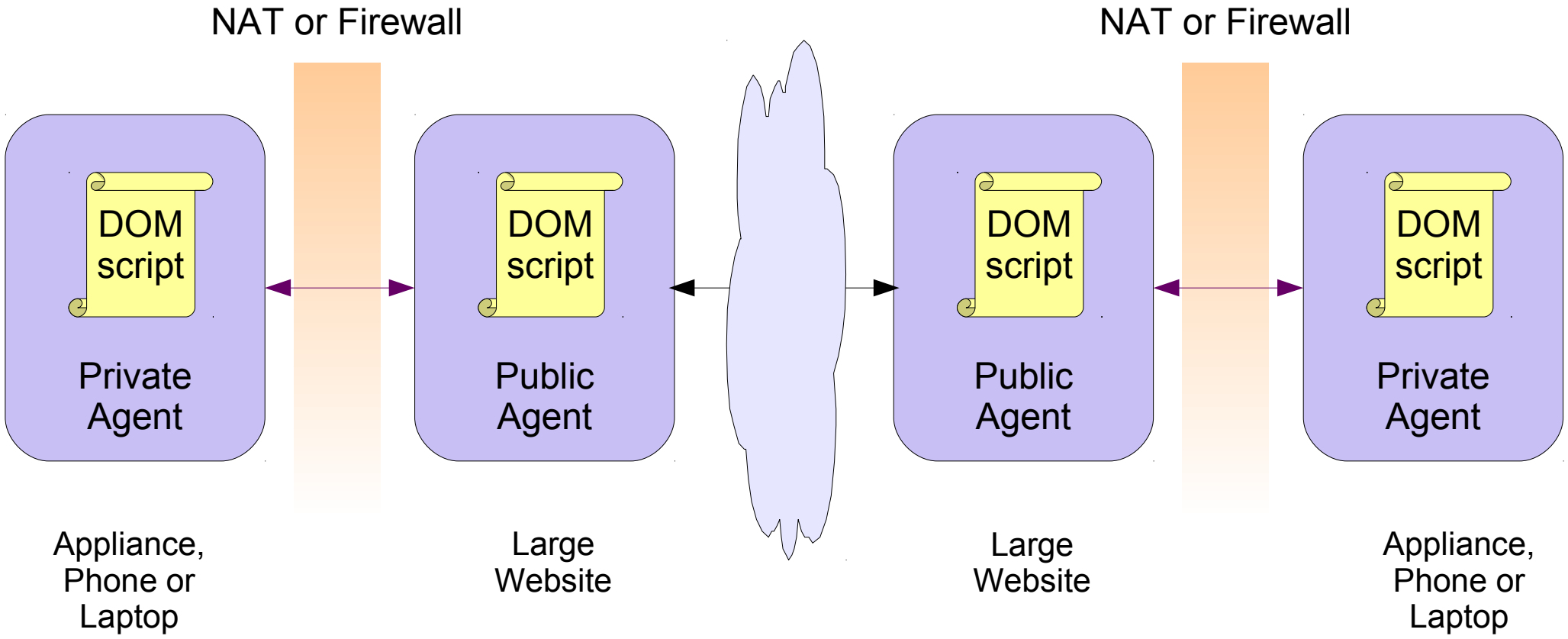
See STUN, TURN and other techniques

Tunnelling through NAT



Connecting devices behind different NATs

Public and Private Agents



- Private agents may be off-line or powered down
- Enabling off-line operation via data synchronization

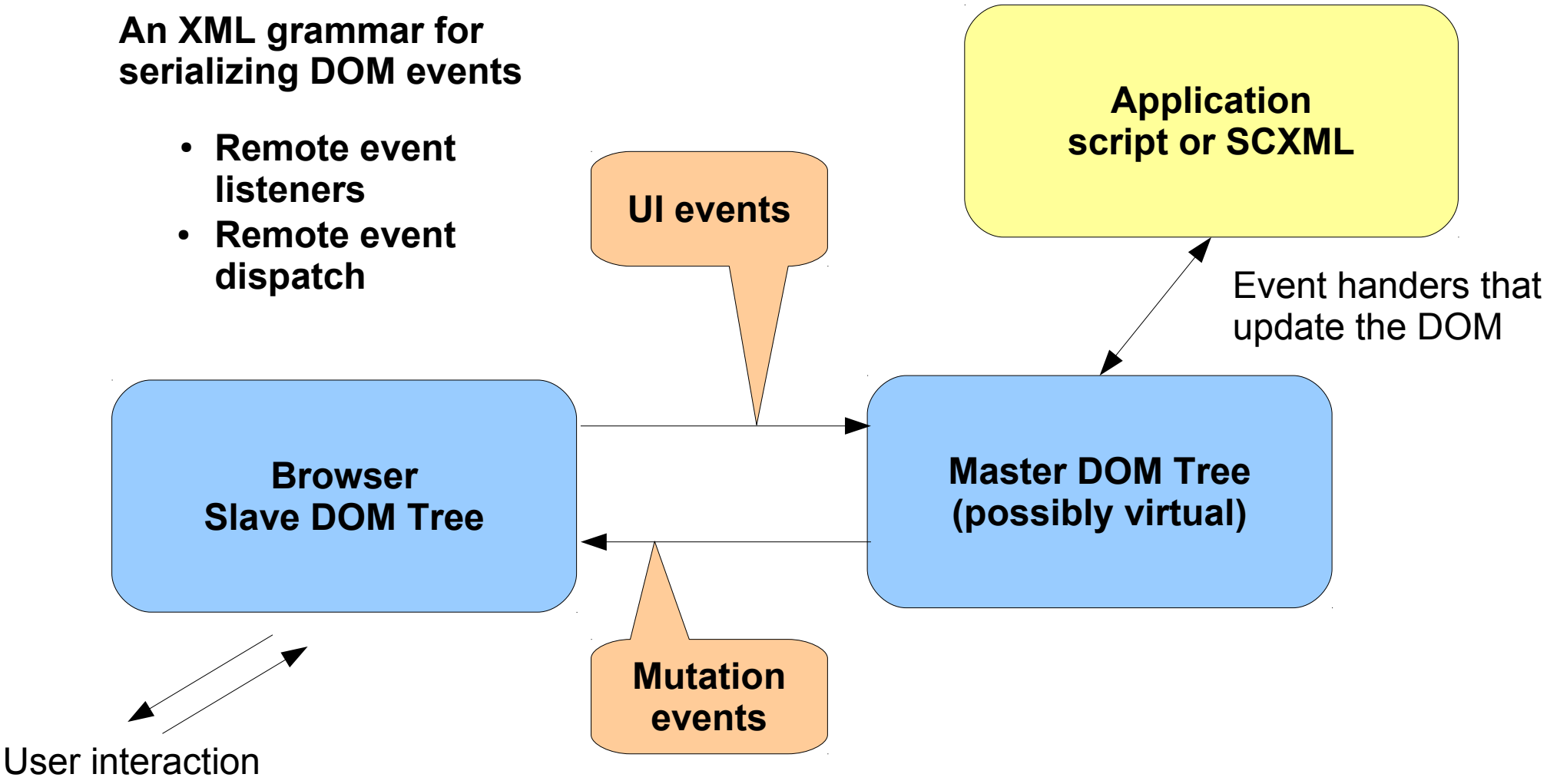
Remote User Interfaces

- Moving beyond Web browsers to new kinds of applications
  - based upon distributed document object models
  - application running on one device is coupled to a user interface on another via an exchange of events
- Layered architecture involving mappings between different levels of abstraction
  - High level events as interpretations of lower level ones
  - Realizing high level tasks as particular UI behaviour

Remote User interfaces

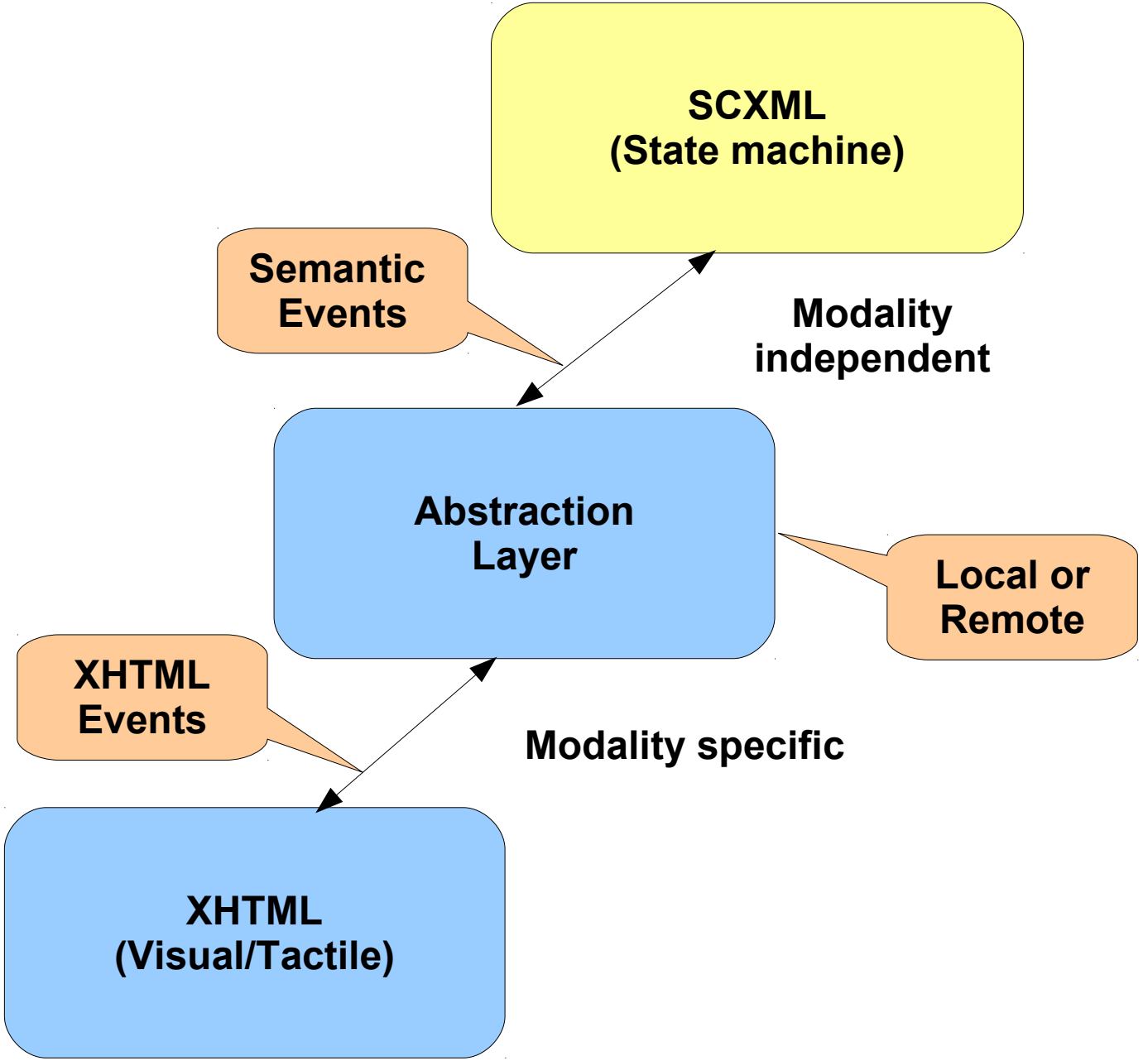
An XML grammar for serializing DOM events

- **Remote event listeners**
- **Remote event dispatch**



DOM = Document Object Model

Abstraction layer for Events



New Directions for Web Authoring

Model-Based UI Incubator Group

http://www.w3.org/2005/Incubator/model-based-ui/

Adaptation

- Describing applications in a way that makes them easier to run on a wide range of devices
- Dynamic adaptation to user preferences, device capabilities and environmental conditions
  - Catering for adaptation at authoring time
  - Server-side use of rich meta-data for adaptation
    - tailor content to match screen, memory, bandwidth, etc.
  - Client-side access to hierarchy of properties and the means to make changes
    - expose battery level within web page UI
    - client side mashup based on access to device location
    - change audio settings from web page UI

Policy-based Adaptation

- Author markup in device independent representation
  - authoring format is freed from browser restrictions
  - high level events in place of low level scripts
- Describe policies for adaptation to classes of devices
  - layout, images, style sheets, scripts, etc.
  - skinning apps as combo of markup, CSS, script
- Adaptation process executes policies for specific delivery context
  - work arounds for variations across browsers
  - split content for low memory devices
  - exploit client APIs for rich web apps (e.g. Ajax)

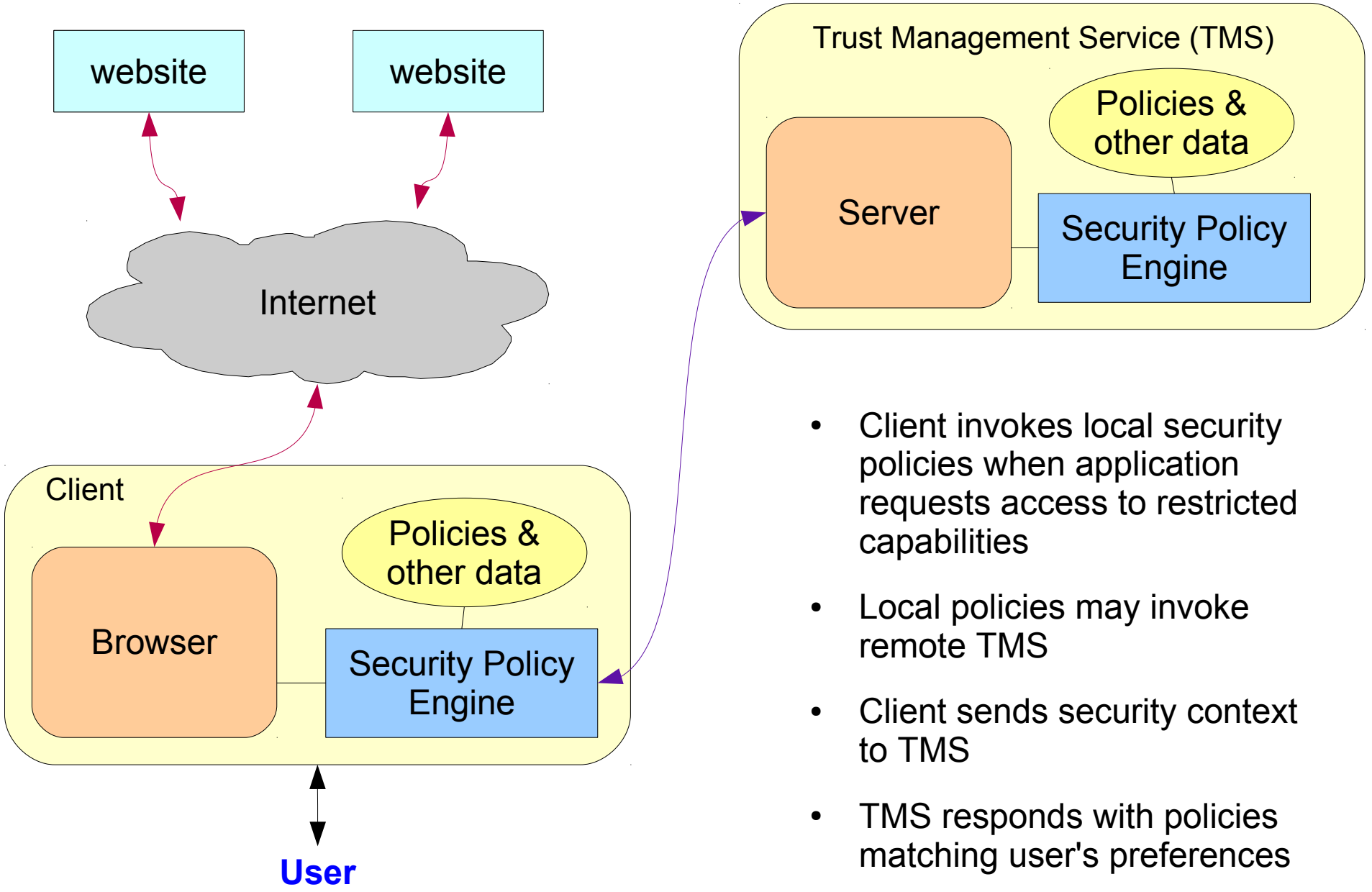
Security and Privacy Concerns

- The Web is a mess when it comes to security
- Different user name/password for each website encourages people to use weak passwords
- Wide open to phishing attacks
- Criminal gangs harnessing compromised PCs to send out spam and to launch attacks
- Privacy abuses are commonplace
- Browser sandbox model and same-site policy are too weak and work-arounds introduce major security/privacy holes

Trust Management Solutions

- Users tend to click through security related dialogues that “get in the way” of the task
- Users are often not really informed about the trustworthiness of a website/application
- We need to find solutions that offer greater security with improved usability
- Improved security through SIM cards and biometric techniques
- New ideas for trust management solutions involving a trusted third party

Trust Management



- Client invokes local security policies when application requests access to restricted capabilities
- Local policies may invoke remote TMS
- Client sends security context to TMS
- TMS responds with policies matching user's preferences

Motivation

- Professional Web applications are developed by teams of people with different roles & skills
- Frequent need for redesign as data models, business requirements and branding changes
- Reduce costs and increase re-use through separation of concerns
- Allows team members to focus on what they each do best
- Outsource tough task of adaptation to particular browsers and devices (analogous to compilers)

Model-based UI Layer Cake

Cameleon Reference Framework

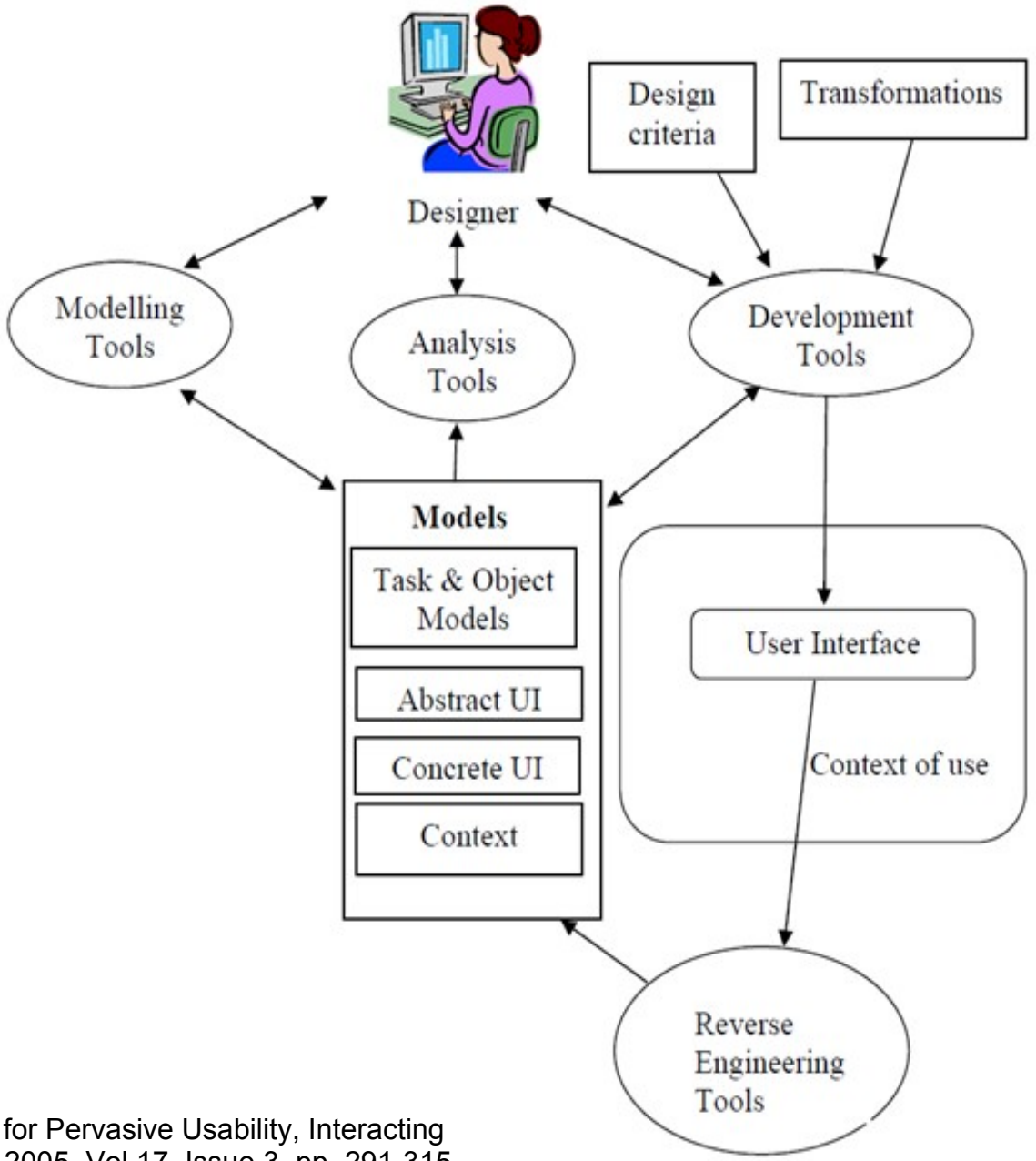
1) Application task and data models
2) Abstract User Interface
   - modality independent, e.g. select 1 from n
   - set size, grouping and ordering considerations
3) Concrete User Interface
   - Commitment to modality and broad class of devices,
     - e.g. radio buttons vs drop-down menu
4) Final User Interface
   - Automatic generation guided by author's preferences
   - Target HTML, SVG, Flash, Java, .Net, etc.
   - Generation of client and server-side components

with transformations defined between each layer

Design Steps

- Start with domain concepts and tasks
  - Arrival and departure dates for a hotel reservation
- What kinds of interaction objects are needed?
  - Selection mechanism for dates
- How do we want to realise these in concrete terms?
  - Pop-up date picker and reservation summary
- What kinds of devices do we want to support?
  - Detailed choices of layout, fonts, colours, and art work

Model-based Development Process



F.Paternò, Model-based Tools for Pervasive Usability, Interacting with Computers, Elsevier, May 2005, Vol.17, Issue 3, pp. 291-315.

What does this mean for authors?

- Authoring tools should hide details of markup
  - Markup languages designed for authoring tools, not for browsers, and not for human editing
  - Focus on separation of concerns, not on brevity
  - Tools that support top-down and bottom up design
- Models held in server-side repositories
  - Enables distributed authoring by team members
- Use of diagrams and rules that are translated into the internal representations of models
  - Much nicer than hacking JavaScript for IE6
  - Painless adaptation to devices and browsers

Model-Based UI Incubator Group

http://www.w3.org/2005/Incubator/model-based-ui/

- W3C Group launched in November 2008
- Mission to study work on model-based UI and see what if anything is ready for standardization
- Participating organizations
  - CNR -- Consiglio Nazionale delle Ricerche
  - Department of Informatics, PUC-Rio
  - Fraunhofer Gesellschaft
  - JustSystems
  - Siemens AG
  - Telefónica de España, SAU
  - Université catholique de Louvain

MBUI XG

- Meets every other week by phone
- Occasional face to face meetings
- Initial charter for 12 months, ending Nov '09
  - Extended until Spring 2010
- Deliverables: Incubator Group Report
  - Survey of existing work
  - Use cases and requirements
  - Suggestions for standardization
- Wiki as basis for joint authoring
  - http://www.w3.org/2005/Incubator/model-based-ui/wiki/Main_Page

MBUI XG Use Cases

- Smart Home Network
  - UI for controlling and monitoring a dynamic network of heterogeneous devices
    - security system, washer/dryer combo, and room fan
- Remote access and control of home devices
  - Did I remember to turn the heating off?
  - Has the neighbour fed the cats?
- Easy development for wide range of devices
  - Accessing services from Desktop, PDA, Phone
  - Rapid prototyping for early user feedback

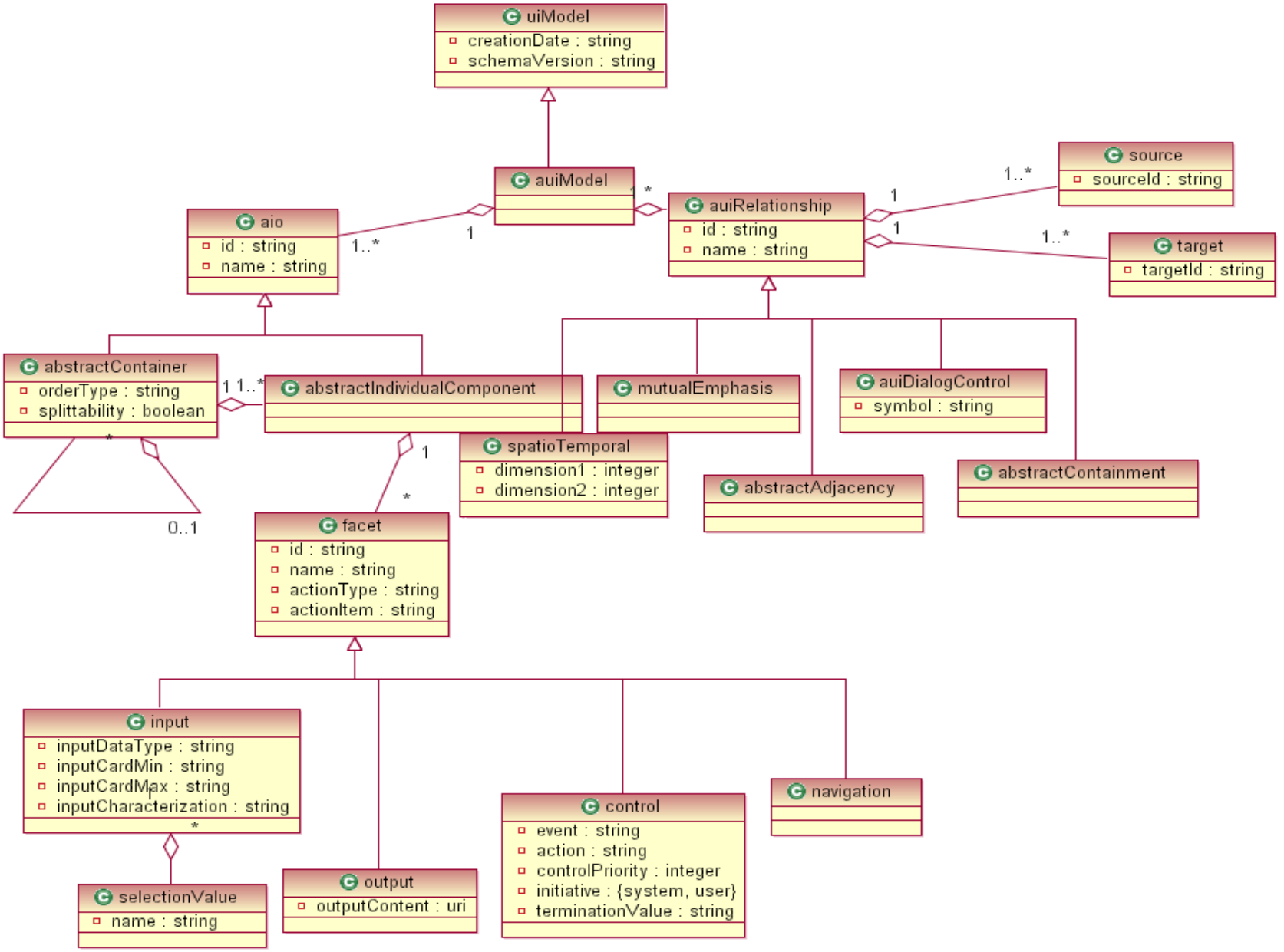
Task Models for UI Design

- Expressible at various abstraction levels
  - High level requirements (task meta models)
    - Concur Task Trees
  - Detailed representation of activities
    - Statecharts, e.g. SCXML
- Some other approaches for task meta-models
  - UsiXML
  - TOOD
  - Diane
  - HTA
  - GOMS

Abstract UI Model

- Interaction at a level independent of modality and device
  - Valuable for creating accessible applications
- UsiXML
  - Guerrero Garcia, J., Vanderdonckt, J. (2008), Towards a Multi-Users Interaction Meta-Model. IAG Working Paper 08/25, Université catholique de Louvain, 2008.
- XForms
  - W3C specifications for Forms
    - Model-View-Controller design pattern
    - Abstract UI controls

UsiXML Meta Model



Concrete UI

- Commitment to specific modalities and broad classes of device capabilities
- W3C WAI/ARIA taxonomy
  - Controls, properties and events
  - Aimed at retrofitting HTML/JS web apps
- UsiXML
- UIML
- Platform specific concrete UI
  - Adobe Flex and MXML
  - Microsoft Silverlight and XAML

Transformations between Layers

- Mappings between objects and events
  - Events as messages exchanged by objects
- Managed by authoring tool
  - Let the machine take the strain
  - Experts can tweak mappings if really needed
- If Layers described in XML, use XSLT, right?
  - Wrong, too powerful to allow machine reasoning
- Principle of reduced power
  - Just sufficient to express what is needed

Relationship to Current Practice

- Tag soup and scripting hell
  - Variations across browsers
  - Ever greater complexity
  - Browser is just the tip of the iceberg
  - Much of the work is on server-side scripts
- Content is locked into specific CMS
- Expensive to deliver content to multiple channels e.g. mobile
- Model-Based approach offer the promise of a way out, eventually ...

Summing up

Service Front-End

- Service front-end as web application
  - Defined as markup and scripts
  - May be local installed as widget
- Exposes UI for
  - Configuring user preferences
  - Browsing context
  - Mashing services
- Exploits context for personalization

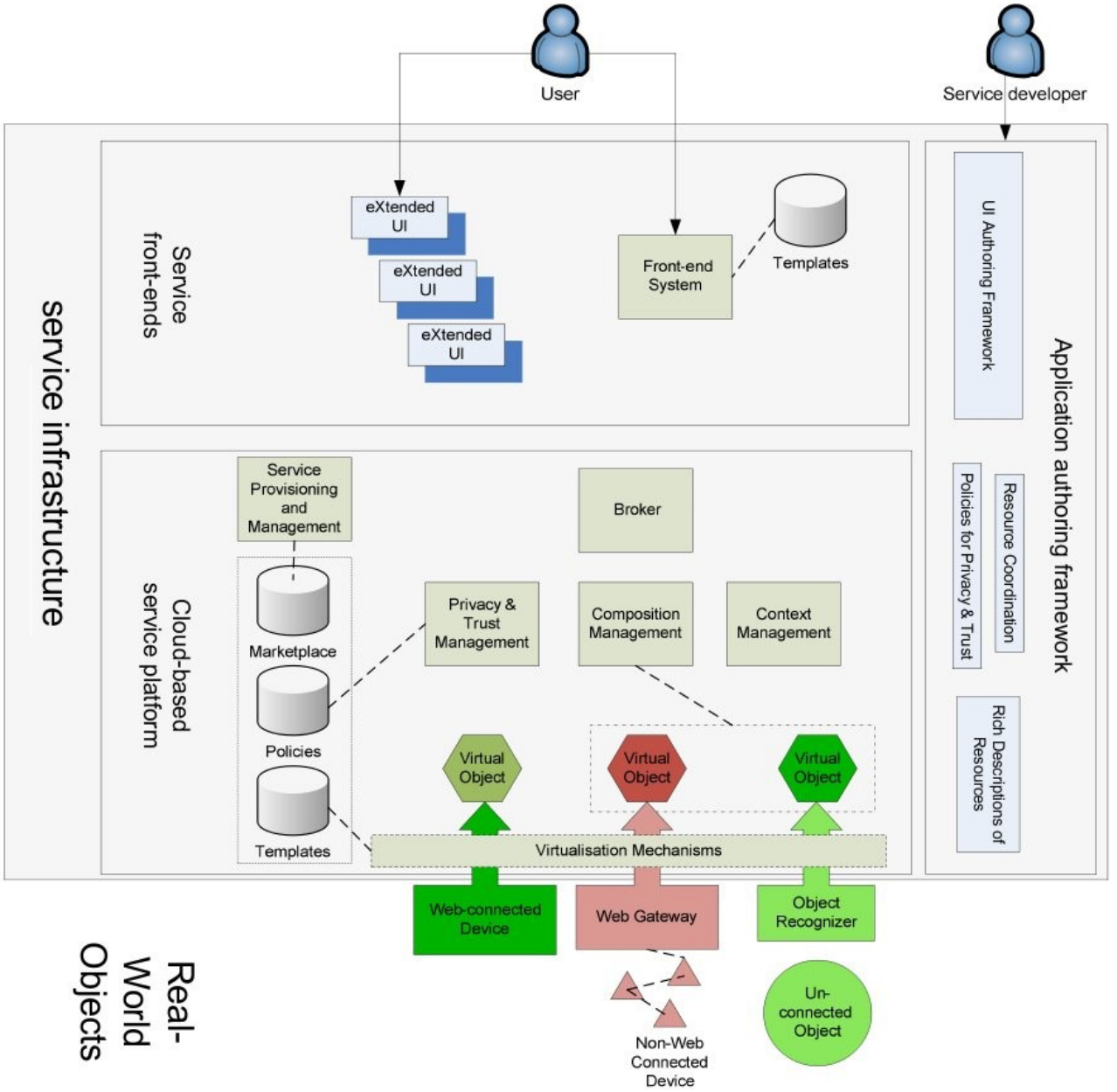
Dynamic Context

- Mechanisms to maintain a dynamic model of the context in the cloud
  - Context as rich description of the world
  - Updated as devices are added or removed
  - Describes what kinds of devices they are
  - How to communicate with each device
- Built on top of existing lower level mechanisms such as UPnP
- May involve a local device as bridge between local protocols and cloud

Web-based Broker

- Web run-time as execution environment for apps expressed as markup/scripts
- APIs that allow apps to query the context
  - Browser app that allows users to explore what devices are present
    - Events that fire when context changes
- Broker that allows apps to bind device or service into the web run-time as proxy objects
  - Objects appear as part of application session
    - Session itself is an object you can query
    - Broker is implicit or named service in cloud

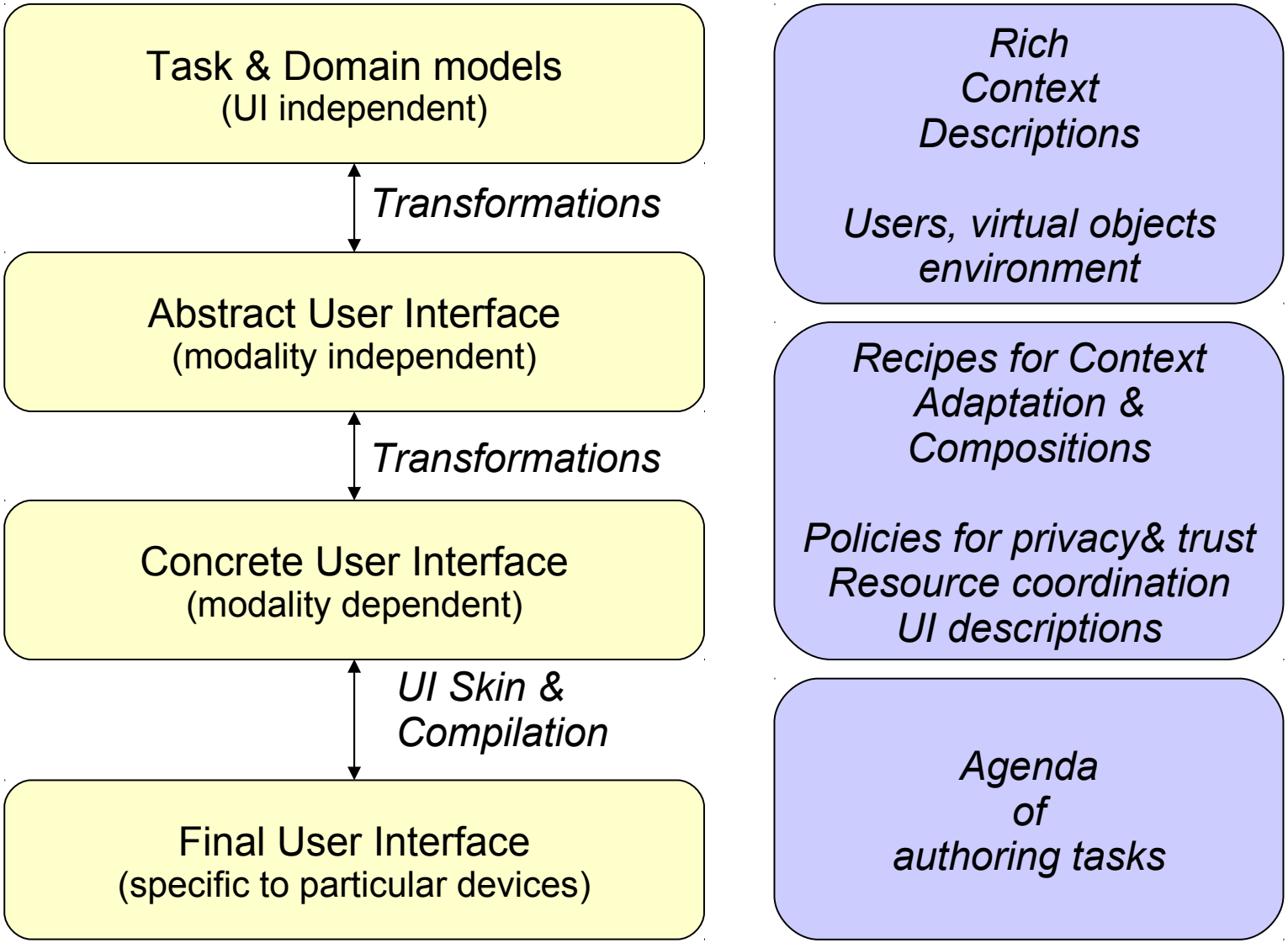
Service Infrastructure



Authoring Framework

- For use by web developers
- Layered representations
  - Separates out different design concerns
  - Uses Cameleon Reference Framework
- XML + event-driven scripts
- Context aware (authoring and run-time)
- Pluggable (tool kit vendors)
- Compiles to delivery platform
  - Guided by author's preferences
  - Deals with platform variations

Authoring Framework



From IOWA Proposal Part B

Ecosystem Layer Cake

Web technologies for connecting users and real world objects for new kinds of services

1. The UI can also use dumb objects that are sensed by devices e.g. cameras
2. A dynamic representation of the context is maintained in the cloud

Summary and Questions

This talk is available at **http://www.w3.org/2010/Talks/0123-dsr-sofsem.pdf**

- The Web of Things
  - The potential for applying Web technologies to distributed applications of all kinds of devices
    - Web-based abstraction layer
    - Role of Semantic Web for rich descriptions
- The Web of Trust and role of delegation
- New directions for Web application authoring
  - Model-based User Interface design