Road to Web 2.0

Eric Burger
Deputy CTO, BEA Systems
Member of the Board, IMS Forum
Member of the Board, SIP Forum

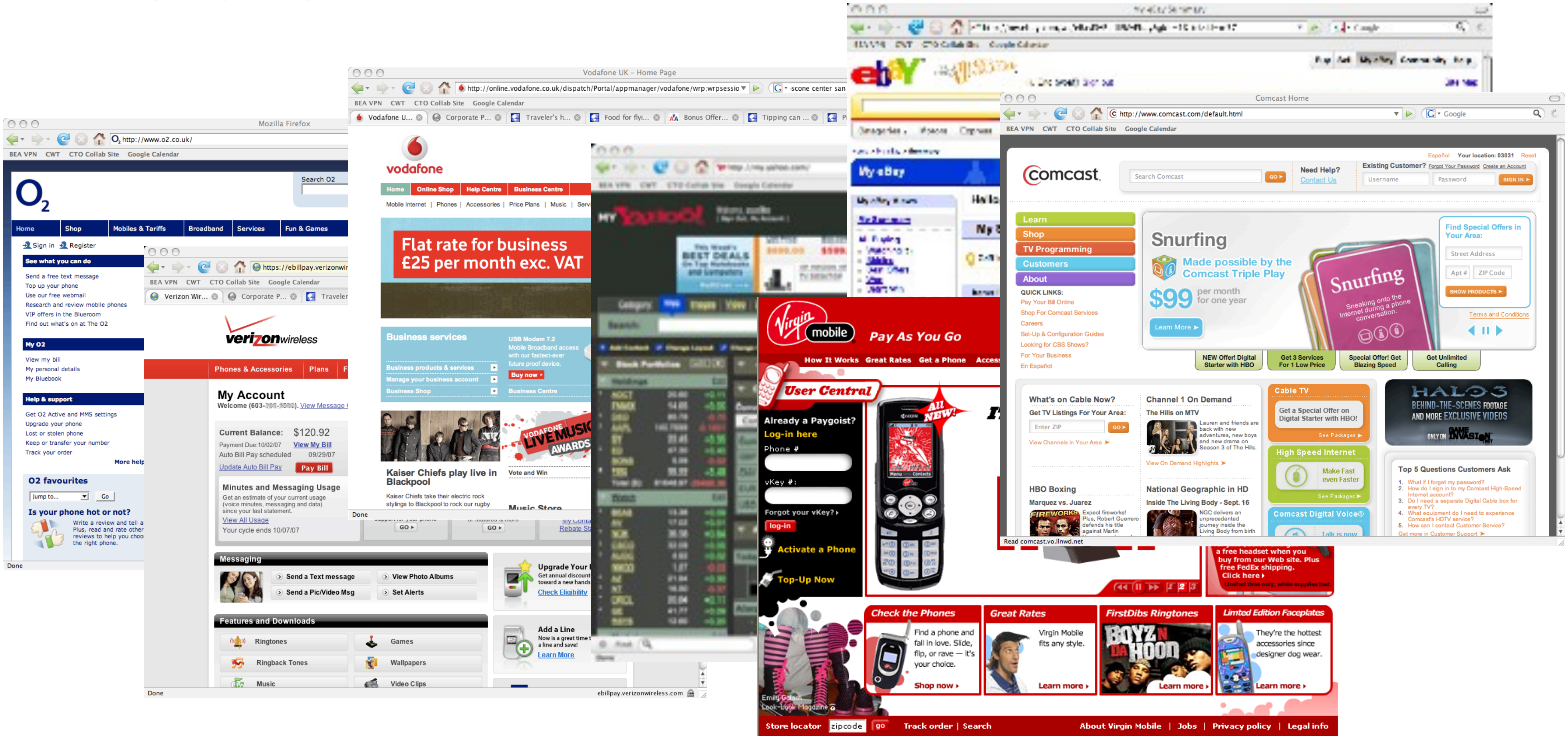
Telestrategies Telco-Web 2.0
September 2007

Think liquid.™

Agenda

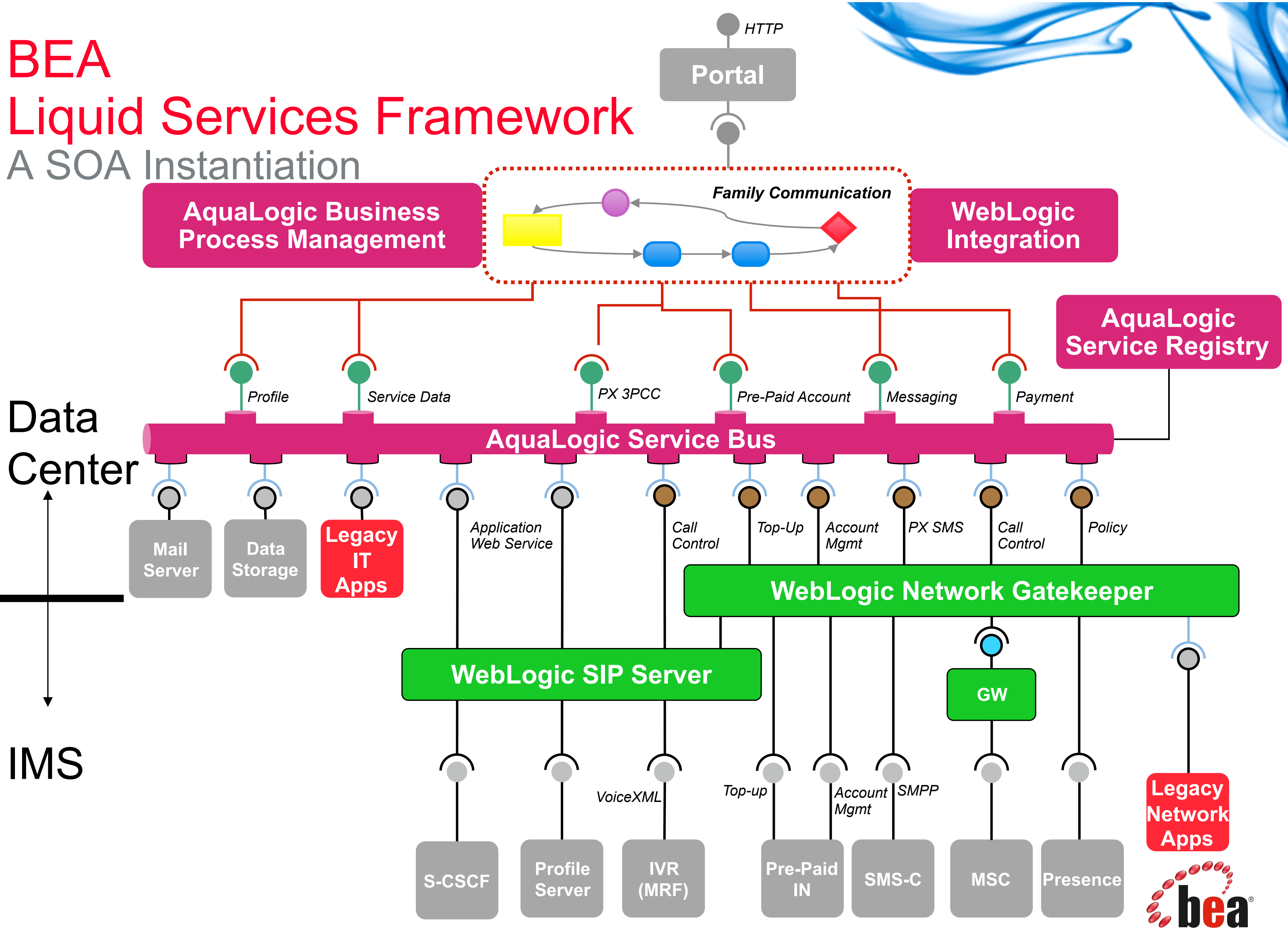
- Who We Are and Why We Are Here
  - What is Web 2.0 and why is it important
  - Who is BEA
- Service Creation and Deployment
- Enablers for User-Generated Applications
- Examples

Where Is BEA?


O2
Vodafone – Home Page
Flat rate for business £25 per month exc. VAT
verizonwireless
My Account
Virgin mobile Pay As You Go
User Central
Comcast Home
Snurfing
$99 per month for one year
ebay

- WebLogic Server is the predominant application server in telecom service providers
- 50 of the top 50 global service providers use BEA technology
- The majority of leading network equipment providers use BEA solutions in their product suites and network deployments

BEA
Liquid Services Framework
A SOA Instantiation



bea® Think liquid.™

Points I'll Pass On (Controversy So Soon?)

- BEA is already #1 in OSS/BSS Infrastructure
  - AquaLogic products for IT (OSS/BSS/CRM)
  - Liquid Services Framework (LSF) for Real-Time Communications Applications
- Formalized SOA to address 12-week / 18-month problem
  - IMS / Stimulus-Markup means applications developed in 12 weeks
  - Still takes 18 months to setup operations, billing, and customer support
- SOA and "Web 2.0" not sufficient
  - Folks not coming from Internet world miss key points of Web 2.0
  - Get lost in blogs, wikis, and mash-ups
    - Great for telco enterprise itself, but not focus of today

Web 1.0: User Generated Content

- Self-published academic papers
- Information sharing (home pages, file sharing)
- Dynamic content (Web applications)

- Popular press forgot roots of web
  - Focus on mass market media as major web application
  - Self-bias?

The Web has always been about
User Generated Content

Web 2.0: User Generated Applications

- Rich environment
- High interaction
- User-directed presentation
- User-integrated applications (mash-up)
- User-generated applications
  - Applets
  - Widgets

How Web 1.0 and 2.0 Applications Differ

- Web 1.0
  - Expert programmers: Java, Python, Perl, C++, etc.
  - J2EE, Containers, POJOs, RubyOnRails, etc.
  - Required ownership of application server or application server host
- Web 2.0
  - **End user** creates applications
  - Final application composition occurs at end user device or proxy
  - Tools and techniques to make it easy enough for
    - Joe Sixpack to create own iGoogle page
    - Jane Semicompetent to create widgets

Web 2.0 Matters

- Purist perspective: embodies original view of the Web
  - Users generating content (current hoopla in popular press)
  - Users generating applications (current hoopla in enterprises)
- UGC (user generated content)
  - Mechanism for collaboration
  - Add communications and get really cool collaboration, business processes, businesses
- UGA (user generated applications)
  - Ecosystem of applications
  - Realistic way to address long-tail of application space

Agenda

- Who We Are and Why We Are Here
- Service Creation and Deployment
  - Myth of the killer application
  - Myth of the expanded service solution space
  - Service Provider perspective
  - A way forward
- Enablers for User-Generated Applications
- Examples

"Low Cost to Fail, Low Cost to Succeed"

-- Eric Burger, VON 2003

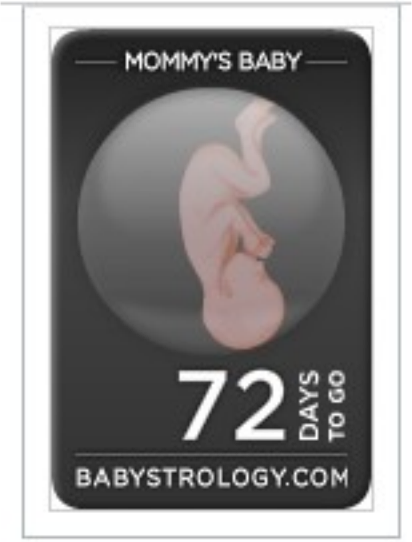
- Promise of stimulus / markup design model
  - Premise of IMS
- So easy to create applications, can explore solution space
- So easy to scale, can deploy if hit the big one

- Enables both focused market research and shot-gun approach
- Providers would be able to reach long-tail of market

- Why do we not see this in practice?

Service Providers / ISV's / NEP's cannot address long-tail

- Almost by definition, SP's are mass-market organizations
- Well-run organizations should focus on corporate focus (mass market)
- No killer application on the horizon
- Promise of IMS was to make it easy to explore solution space to discover killer application
- #2 Web Widget is?



http://www.widgetbox.com/widget/baby-ticker

Who Will Create the Applications?

- Communication service provider culture
  - Labs: prototypes
  - Network engineering: standards, vendors
  - IT
    - Deliver products, but real-time interactive multimedia?
    - Clash with Network?
- ISV culture
  - Network-focused
  - Well run: focus on customer requests

The Users of the Network Create Applications

- Many may look like (become) companies in the end
- Many will be companies
- Many will be enterprises: SOX, HIPPA, BASEL II

The interesting application is:

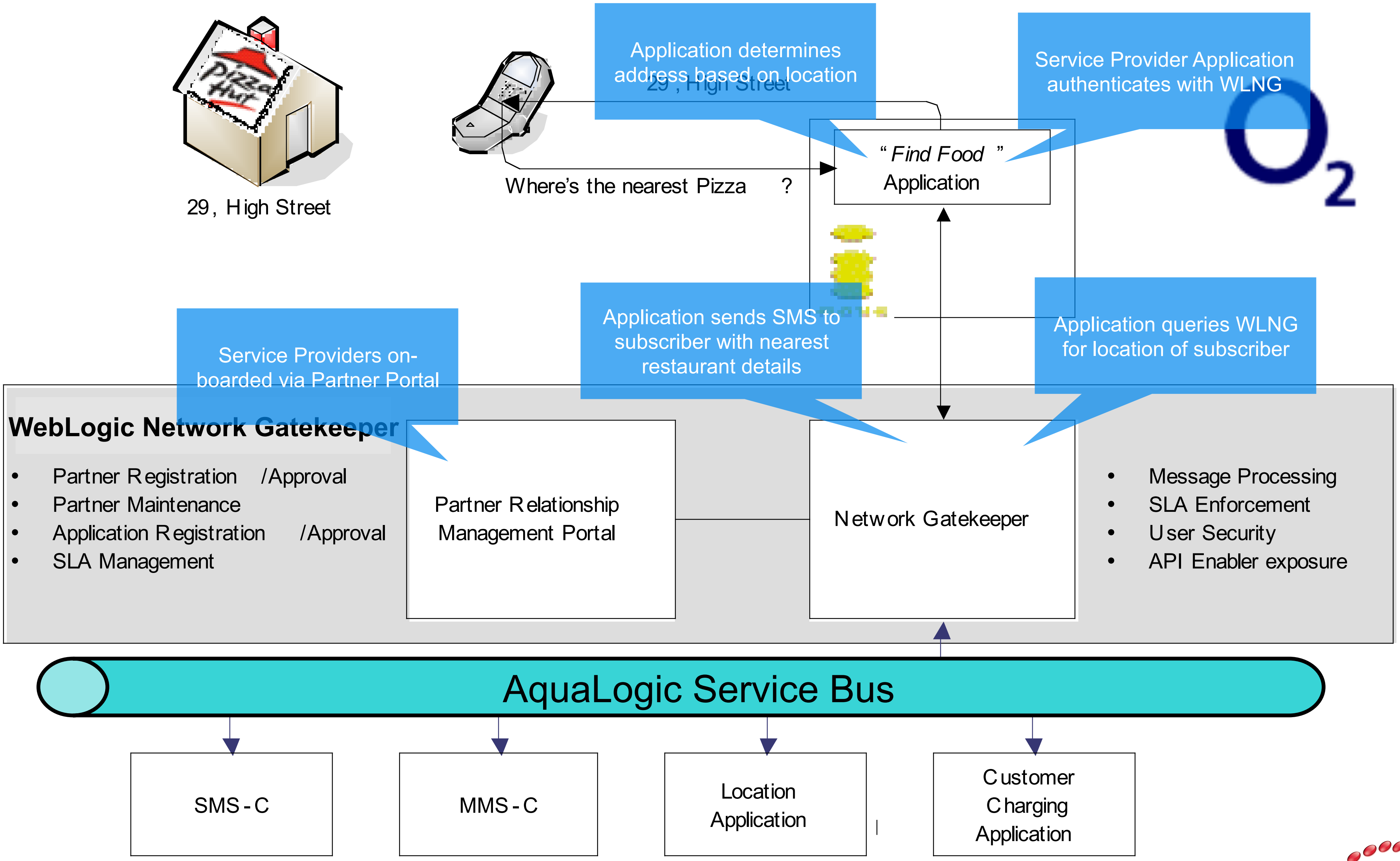
Ability of **user** to create their **own** application, to **use** or **share**

- What if we let ANYONE create applications?
  - What do we need to have to enable this?

Agenda

- Who We Are and Why We Are Here
- Service Creation and Deployment
- Enablers for User-Generated Applications
  - Service Exposure
  - Policy Enforcement
  - Security
  - Business Model
- Examples

Is This Web 2.0?



29, High Street

Where's the nearest Pizza ?

Application determines address based on location

29, High Street

Service Provider Application authenticates with WLNG

"Find Food" Application

Service Providers on-boarded via Partner Portal

Application sends SMS to subscriber with nearest restaurant details

Application queries WLNG for location of subscriber

WebLogic Network Gatekeeper

- Partner Registration /Approval
- Partner Maintenance
- Application Registration /Approval
- SLA Management

Partner Relationship Management Portal

Network Gatekeeper

- Message Processing
- SLA Enforcement
- User Security
- API Enabler exposure

AquaLogic Service Bus

SMS-C

MMS-C

Location Application

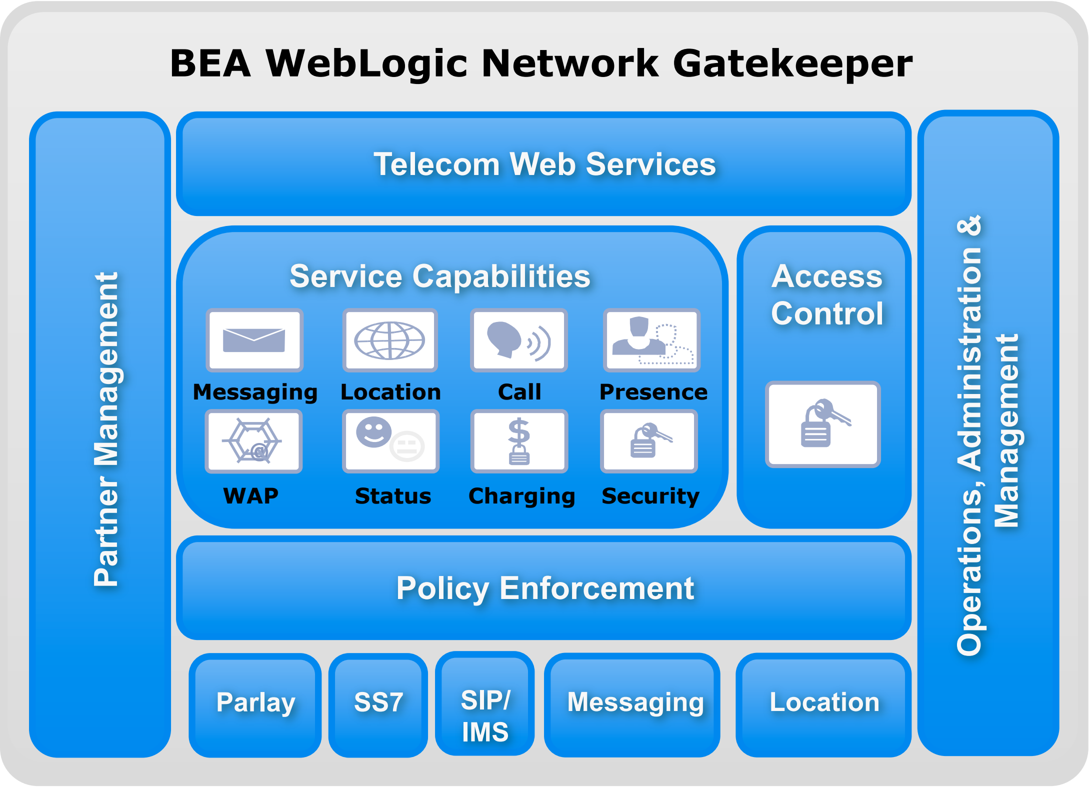
Customer Charging Application

User-Generated Applications are Not Trusted

- Web 2.0 is not about
  - Service providers (self-trust) building their own applications
    - Although it greatly helps development (0.5 day versus 9 months)
    - Although it greatly helps development process (collaboration)
  - Integration of silo applications into OSS/BSS (self-trust)
    - Although SOA helps greatly here
  - Trusted ISV's building semi-custom applications for service providers
    - Although it greatly helps development
- Web 2.0 is about
  - Enterprise customers (only contractually trusted) having control of their applications and data
  - Users (not trusted at all) able to create their own applications

Service exposure

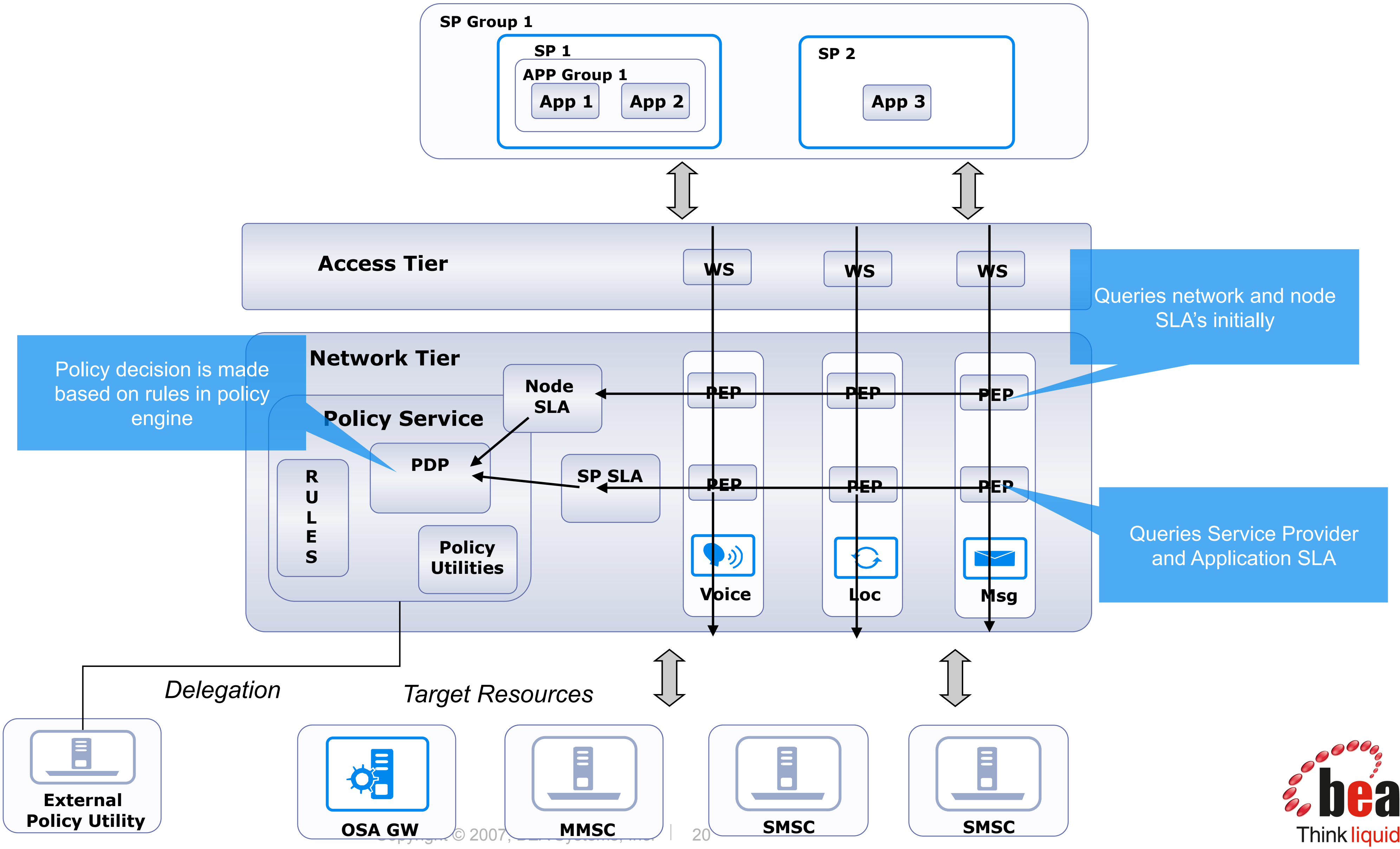
- Becoming table stakes
- ParlayX
- Billing Services
- Advanced Web Services
- Discussed ad nauseum in other sessions



Policy Key to Open Networks

- Policy: runtime execution of rules
  - SLA between network operator and service provider or user
  - Operator usage characteristics
- Enforcement types
  - Application SLA: Limit per application, service provider / group
  - Service feature protection
  - Traffic management SLA: Routing traffic to specific network resources

Policy Enforcement



Queries network and node SLA’s initially

Policy decision is made based on rules in policy engine

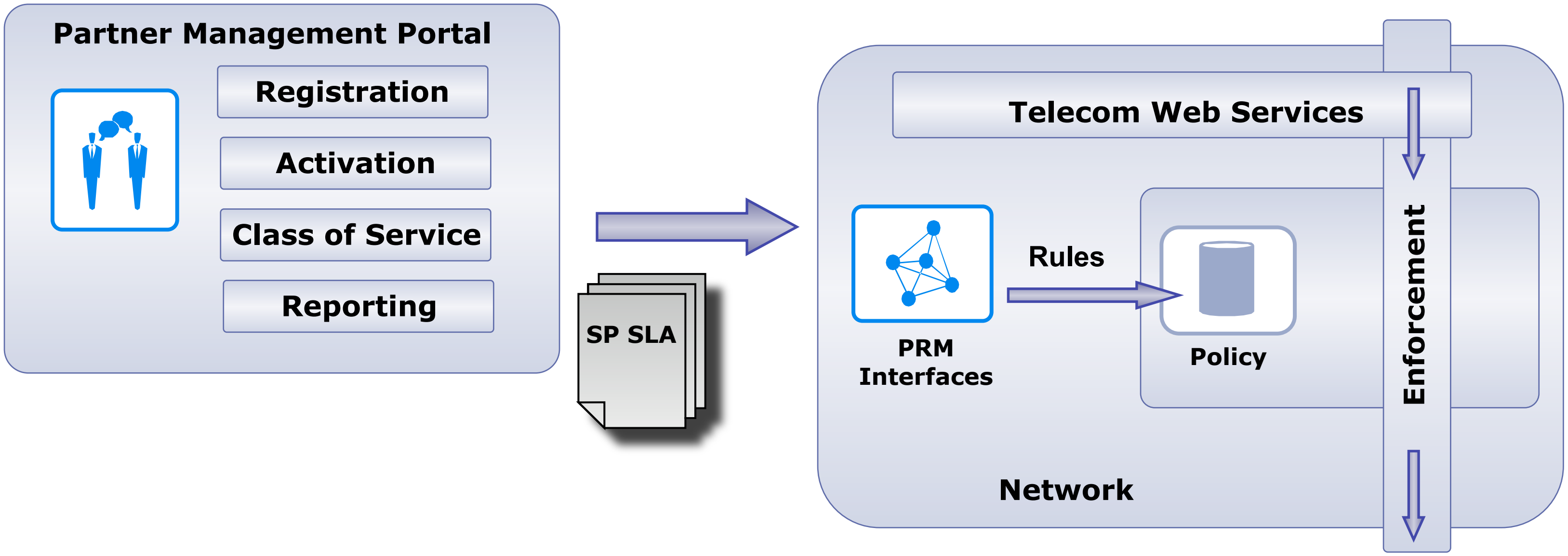
Queries Service Provider and Application SLA

Carriers Who Expect to Get Paid Expect…

- Authentication
  - You are who you are, really
  - I prove I am who I am, really
  - Username/Password, Certificate, Digest, SSO, Shared Hash
- Security
  - Authentication demonstrably secure and immutable
  - User data protected
  - Operator network and data protected
  - WS-Security, WS-Policy, WS-Trust, WS-SecureConversation
  - ACL, TLS, S/MIME, SAML
- Audit
  - Who did what, when, to whom

Provisioning: Partner Portal

- Partner Portal is central point for on-boarding partners
  - The partner data needs to be integrated with system in order to perform SLA enforcement on behalf of those partners
  - Out-of-the-Box policies and the ability to customize policies to enforce SLA's are required



Business Model

- Can be anything
  - Ad-supported (culture?)
  - Subscription to end users (who are the end users?)
  - Subscription to service providers (by whom?)
  - Free (cost saving on churn or customer acquisition)
- Service provider subscription
  - Clear $$/month, standard telco model
  - Include churn reduction (late 1990's model)
  - Include drag (2000's model)

Summary

Web 2.0 is about UGC and UGA

CSP’s cannot continue with “business as usual”

CSP’s and ISV’s cannot structurally address market

Opening network requires policy and security enforcement

Killer application is the environment

Thank You

Eric Burger
eburger@bea.com

bea
Think liquid.™