SCO Forum 2006

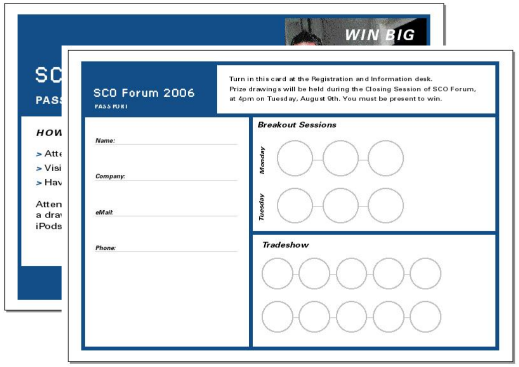
MOBILITY EVERYWHERE ›

Presentation Title: Building Applications for the EdgeClick Framework
Presenter’s Name: Jonathan Schilling
Session ID: 105

Get Your Passport Stamped

- Be sure to get your Passport stamped.
  - Get your passport stamped
    - By breakout session instructors
    - By exhibitors in the exhibit hall
  - Turn in your Passport
    - After the last breakout session on Wednesday
    - Drawing for great prizes for Wrap-up Session

- Remember to complete the breakout session evaluation form, too


WIN BIG
SC
PASS
HOW
> Atte
> Visi
> Hav
Atten
a dra
iPods

SCO Forum 2006
PASSPORT

Turn in this card at the Registration and Information desk.
Prize drawings will be held during the Closing Session of SCO Forum, at 4pm on Tuesday, August 9th. You must be present to win.

Name:

Company:

eMail:

Phone:

Breakout Sessions

Monday ○ ○ ○

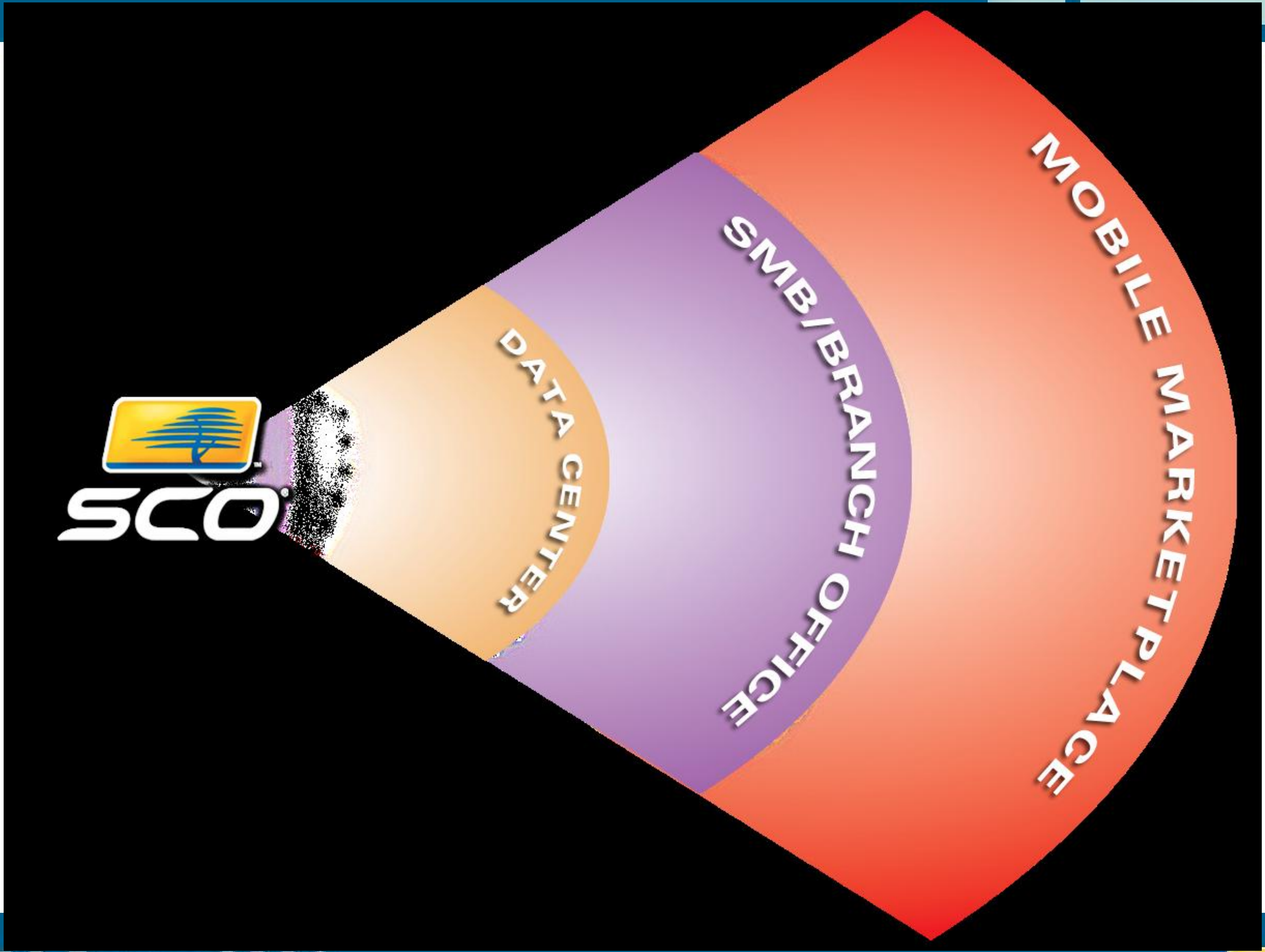
Tuesday ○ ○ ○

Tradeshow

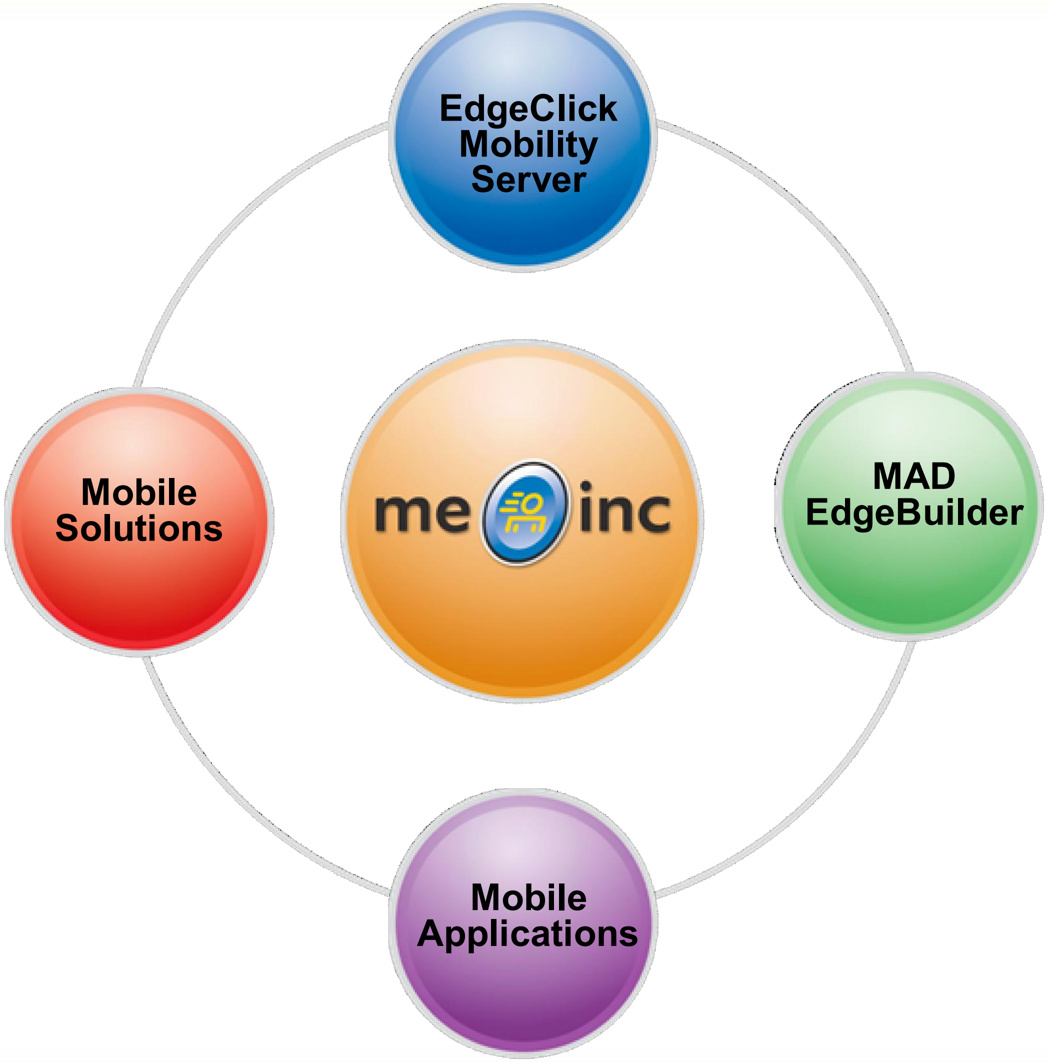
○ ○ ○ ○ ○

○ ○ ○ ○ ○

SCO Automates Transactions



Me Inc. – Mobility Everywhere



Goal

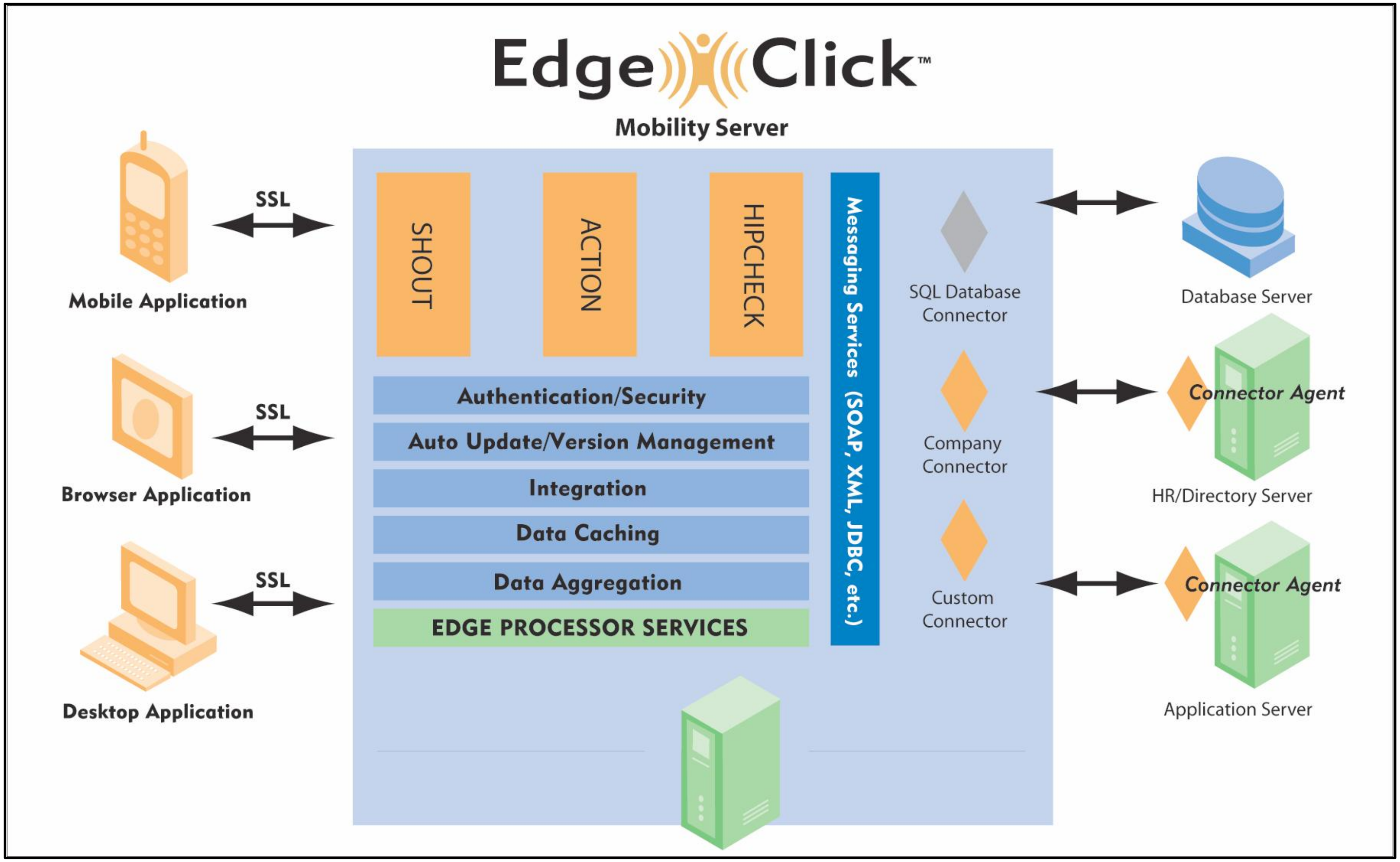
- This is an introduction to EdgeClick development
- “The Big Picture”
- Covers types of EdgeClick applications and how to approach designing them
- Mentions other topics which other breakout sessions will handle in more depth

EdgeClick™

Contents

- Components of an EdgeClick mobile digital service
- Different types of EdgeClick mobile digital services
- How to design EdgeClick mobile digital services
- Technologies used in EdgeClick mobile digital services

EdgeClick Architecture



EdgeClick Architecture Components

- Client part of service
  - Smartphones, PDAs, some regular phones
  - Web browsers
  - Native PC desktops
- EdgeClick Processor (Mobility Server) part of service
  - Service processor
  - Web admin pages
  - Local database
- Back-end integration part of service
  - Connectors
  - Agents

EdgeClick Architecture Tools

- EdgeBuilder
  - The SDK for EdgeClick
    - a/k/a “MAD”
  - Provides libraries of reusable building blocks
  - Provides plug-ins and templates for popular IDEs
    - Visual Studio for Mobile
    - NetBeans
    - Eclipse
    - Sun Wireless Toolkit

Design questions about the architecture

- These are a lot of components
  - Are all of them always necessary?
  - Which are the “primary” components?
- Where to start when creating a new service?
- Which tools to use?

Types of EdgeClick Mobile Digital Services

- Self-Contained Service
- Business Integration Service
- Monitoring Service
- Mobile Database Service

Self-Contained Service

- A new application
  - Does not connect to existing application
  - No agents necessary
- Subscriber-centric
  - Also groups, contacts, etc.
- Most logic on EdgeClick Processor
- Stores application data in ECP database
- Will often not need web admin pages
- Can be tied to “landing pages”
- Examples: Shout, Vote, Action, MIGS

How to design a self-contained service

- Understand EdgeClick *ITeam* semantics
  - Accounts, domains, subscribers, groups, contacts, etc.
- Design EP database schema
- Design client-to-EP flow
  - UI forms on client
  - Client-to-EP commands/parameters interface
  - XML returned to client

Business Integration Service

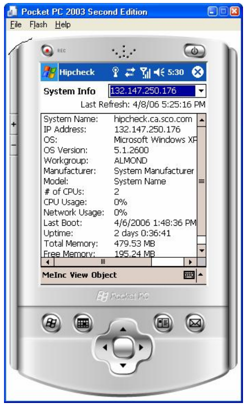
- Connects to existing application
  - Therefore, has connectors or agents
- Purpose is data collection, data reporting, transaction processing
- EdgeClick Processor service is mostly a “switch”
  - ECP database mostly just used for routing
- Application is client- and agent-centric
- Has web admin pages to manage agents
- Examples: Musco Foods, Carlamobile, Meter

How to design a business integration service

- Decide what existing application functionality will be exposed to mobile device
- Design client-to-agent flow
  - The ECP is really just a relay switch here
- Determine where agent should go and how it should be written
- Design small ECP database schema
  - Identify/authenticate subscribers in app terms
  - Direct subscribers to appropriate agents

Monitoring Service

- Specialized variant of business integration service
- Tracks vital business and system metrics
  - View health indicators of business or system
  - Receive alerts based on user-set triggers
  - Take actions to correct problems
- Combines SOA with EDA
- Examples: HipCheck (for Operating Systems)



System Info 132.147.250.176

Last Refresh: 4/8/06 5:25:16 PM

System Name: hipcheck.ca.sco.com
IP Address: 132.147.250.176
OS: Microsoft Windows XP
OS Version: 5.1.2600
Workgroup: ALMOND
Manufacturer: System Manufacturer
Model: System Name
# of CPUs: 2
CPU Usage: 0%
Network Usage: 0%
Last Boot: 4/6/2006 1:48:36 PM
Uptime: 2 days 0:36:41
Total Memory: 479.53 MB
Free Memory: 195.24 MB

How to design a monitoring service

- Imitate HipCheck!
- Design agent services first
- Design client UI to watch/set alerts/take actions
- EP service is a straightforward relay
- Web admin pages also follow straightforwardly
- Coming – "Generic Heartbeat", which allows building monitoring services rapidly by just coding agents to a specified interface

Mobile Database Service

- Keep full application database on mobile device
- Periodically synch with back-end server
- Very client-centric
- Examples: “Big Indian Government Project” – mobile local census info

How to design a mobile database service

- Select mobile device that meets requirements
  - Memory card size
  - Media capture capabilities
  - Programming technology good at files, database
- Service is clearly client-centric
- EdgeClick Processor role may be minimal

Variants on Mobile Database Service

- All sorts of data capture are possible
- Biometric input devices with generic interfaces
  - e.g. fingerprint scan
- All sorts of synchronization protocols with ECP and/or back-end servers are possible
- With immediate relay, resembles business integration service

Client Development Environments: EdgeBuilder SDK

- Windows Mobile .NET/C#
  - Four OS subvariants
  - Treo 700, Samsung 730i, …
  - Visual Studio for Mobile - best dev env and functionality
    - EdgeBuilder template
  - EdgeBuilder SDK provides library for account mgmt, ECP communication, etc.
  - Client apps will also run on PC Windows with some rearrangement

- Java Micro Edition
  - Many variants (even in MIDP 2.0)
  - Many phones, many phone OSes
  - Portability can be difficult to achieve
  - Subset functionality for cross-phone support
  - EdgeBuilder SDK provides library for account mgmt, ECP communication, etc.

Client Development Environments: Other

- HB++
  - Treo 650 and PalmOS
  - Best-looking UI
  - Programming can be difficult
- SuperWaba
  - Java variant for high-end phones
  - Good-looking UI, good portability
  - Little known, support questionable
- AJAX
  - Conventional web browsers
  - cf. WebFace, Google
- Others
  - C++ variants
    - Symbian, BREW
  - Python
  - Etc.
  - Anything that can do http name/value pairs can be used, but no direct EdgeBuilder SDK support …

EdgeClick Processor Service Development Environment: EdgeBuilder SDK

- OS neutral
  - EdgeBuilder currently on Windows XP
- Java based
- Service processor
  - Uses Java app server infrastructure
    - Was Tomcat, now JBoss, later who knows
    - Should be almost completely transparent to application code
      - Java custom annotations used to isolate infrastructure dependencies
    - Don’t have to be a general J2EE expert!
- Web admin pages
  - Spring with Tapestry
  - More flexibility coming …
- “Commons” libraries
  - Data object layer, database common to both service and admin
  - Use whatever you like … EJB3 Persistence, Spring/Ibatis, native Ibatis, JDBC …

Technologies for Connectors

- Connectors run on EdgeClick Processor
  - Communicate directly with existing back-end system
  - Use for pulling in subscriber/group info from HR DB
    - Alternate *ITeam*/*IAccount* source
  - Some applications or database products provide direct web service access
  - Use direct WS, JCA, JDBC, ftp … may avoid agents
  - Firewall issues
  - Solution will be unique to each customer/partner/application/technology

Technologies for Agents

- Agents run on an existing back-end system
  - Agent access is by SOAP/XML-based web services
    - Supported by many OSes, languages, tools
    - SCO Unix provides in Java, C, C++, Perl, PHP
  - Expose apps as services, write database code as service, construct intermediary flat file or mini-db …
  - Firewall issues
  - Solution will be unique to each customer/partner/application/technology

Why not …
use phone browser as the client?

- Characteristics of phone browsers
  - Tend to poor presentation layout
  - Can be unreliable
    - Treo 650 Blazer, for example
  - Can be slow
    - Full page refreshes, no state
  - Difficulty in accessing phone features
    - Camera, fingerprint scanner, voice recorder, etc.
  - Difficulty in maintaining local data
    - EdgeClick supports local data, encryption, store-and-forward

Why not …
go directly from client to back-end server?

- This can be done as a one-off …
- But loses many services that EdgeClick provides
  - Common user, group & account model
  - Common authentication
  - Common https transaction processing
  - Common data caching, store-and-forward
  - Common load balancing, scalability, reliability
  - Common connectors and agents framework
  - (Optional) subscription-based billing model
  - Ability to have apps invoke other apps as services

Summary

- EdgeClick supports building several different types of mobile digital services
- Same tools, libraries, methods, and deployment platform are used for all of them
- Superior to other approaches for mobilizing applications
- Visit other EdgeClick breakout sessions to learn more about each component

Q & A