A System for *In Situ* Tracing and Capturing of User Feedback on Mobile Phones

Jon Froehlich

Mike Chen², Sunny Consolvo², Beverly Harrison², and James Landay¹,²
Mobile devices are used in a variety of contexts
For example, past research has looked at translating lab-based methods into a “mobile setting”
goal

- Create a software tool that collects data about *real* device usage & context *in the field*

- Data can be used to
  - Better understand actual device/system usage
    - E.g., how mobility patterns affect access to WiFi
  - Inform the design of future systems
    - E.g., optimize battery utilization algorithms based on charging behaviors
research challenges

1. **Coverage**: collect rich information about features of interest

2. **Scale**: collect large amounts of data over long periods of time

3. **Extensible**: easily add new data collecting capabilities

4. **Situated**: collect *real* usage data in its natural setting

5. **Robustness**: protect or backup data collected in the field
MyExperience combines automatic sensor data traces with contextualized self-report to assist in the design and evaluation of mobile technology.

Device usage and environment state are automatically sensed and logged.

- Technique scales well
- Cannot capture user intention, perception, or reasoning

Users respond to short context-triggered surveys on their mobile device.

- Can gather otherwise imperceptible data
- Lower sampling rate than sensors
sensors, triggers, actions

Sensors

Example Sensor:
- DeviceIdleSensor
- PhoneCallSensor
- RawGpsSensor
- PlaceSensor
- WiFiSensor

Triggers

Example Triggers:
- DeviceIdle > 15 mins
- PhoneCall.Outgoing == true
- Gps.Longitude == "N141.23"
- Place.State == "Home"
- WiFi.State == "Connected"

Actions

Example Actions:
- SurveyAction
- ScreenshotAction
- VibrationAction
- SmsSendAction
- DatabaseSyncAction
- **XML : Declarative**
  - Define sensors, triggers, actions, and user interface
  - Set properties
  - Hook up events

- **Script : Procedural**
  - Create fully dynamic behaviors between elements specified in XML
  - Interpreted in real time
  - New scripts can be loaded on the fly

```xml
<sensor name="Place" type="PlaceSensor">
  <prop name="PollInterval">00:00:01</prop>
</sensor>

<trigger name="Silent" type="Trigger">
  <script>
    placeSensor = GetSensor("Place");
    if(placeSensor.State == "Work"){
      ... do some action ... 
    }
  </script>
</sensor>
```
We would like to build a model of phone profile behavior (i.e., setting the phone to silent)

Can we begin to predict phone profiles based on sensed context?
  - Time of day
  - Location
  - Transportation mode
  - Calendar appointments
<sensors>
  <sensor name="PhoneProfileSensor" type="PhoneProfileSensor"/>
  <sensor name="RawGpsSensor" type="RawGpsSensor"/>
  <sensor name="CalendarSensor" type="CalendarSensor"/>
  <sensor name="MobilitySensor" type="MobilitySensor"/>
</sensors>

<actions>
  <action name="PhoneProfileSurvey" type="SurveyAction">
    <property name="EntryQuestionId" value="PhoneProfileReason"/>
    <property name="TimeOutInterval" value="00:05:00"/>
  </action>
</actions>

<triggers>
  <trigger name="PhoneProfileTrigger" type="Trigger">
    <script>
      profileSensor = GetSensor("PhoneProfileSensor");
      if(profileSensor.StateEntered = "Silence" and GetRandom() < 0.2){
        GetAction("PhoneProfileSurvey").Run();
      }
    </script>
  </trigger>
</triggers>
architecture

**MEDIA CAPTURE**
- Wireless
- Database Sync
- Self-Report Survey
- Send SMS / Email

**TRIGGERS**
- Calendar.Status == Busy
- Sms.SentMessage == true
- Device.IsIdle == true
- Phone.IncomingCall == true
- Activity.Duration("Running") > 20 min
- Camera.NewPhotoCaptured == true
- Location == "Gym"

**SENSORS**
- Location Sensors
- Bluetooth Sensors
- User Interaction
- Device State

**ACTIVE ACTIONS**
- Media Capture
- Database Sync
- Self-Report Survey
- Send SMS / Email

**CORE**
- Scripting Engine
- XML
- SQL Server 2005 Mobile
- DATA LOGGER
- LOG.TXT
- Object Factories

**DEVICE BACKEND**
- SQL Server2005 Workstation
- HTML

**WEB SERVER**
Windows Mobile 5
- .NET CF 2 in C#
- SQL Server Mobile 2005
- A few open source libraries
- Port of Simkin

**Component** | **Lines of Code**
--- | ---
MyExperience | 16,970
Roam | 14,271
SimkinCS* | 9,148
**Total** | **40,389**

*SimkinCS is a port of Simkin, a java-based scripting language*
performance

HTC Tornado SmartPhone

HTC Universal Pocket PC Phone
Current build of MyExperience is 1.56 MB
- Includes ~150 sensors (e.g., Sms, Phone, GSM)
- 11 actions (e.g., Surveys, Database Sync)

Phone must also have SQL Server Mobile 2005 and .NET CF 2
- Windows Mobile 6 comes with this installed

Memory footprint
- 4.32 MB of memory (< 20% available on most devices)
battery life

- **Baseline**
  - 4 days, 17 hours

- **With MyExperience**
  - 140 active sensors
  - 20 survey actions / day
  - 4 days, 3 hours (~12% decrease)

- **WiFi, Bluetooth based sensors** will decrease battery life substantially (~50%)
< 3% utilization at a rate of 4320 actions / day
sensor performance

- It takes approximately 1ms for a sensor state change to propagate through system and be stored in local database
- Sensors that fire rapidly ~1,000 Hz can starve the CPU
  - We designed our sensors to run at 1 – 10 Hz
  - GPS sensor and mobility sensor run at 1Hz
  - Accelerometer-based sensor runs at 4Hz
case study 1: charging behavior

- **Motivation**
  - Battery life has long been a challenge in mobile computing
  - Dependent on usage:
    - WiFi, video, length of calls

- **Study**
  - 2 week pilot study with 4 people
  - Log device usage (e.g., phone calls, WiFi, active applications)
  - Actively track battery life
  - Survey at moments of charging
<sensors>
  <sensor name="SystemStatesSensor" type="SystemStatesSensor"/>
  <sensor name="BatteryLifeSensor" type="BatteryLifeSensor"/>
  <sensor name="PowerChargingSensor" type="PowerChargingSensor"/>
</sensors>

<actions>
  <action name="BatteryLifeSurvey" type="SurveyAction">
    <property name="EntryQuestionId" value="CurrentLocation"/>
  </action>
</actions>

<triggers>
  <trigger name="BatteryLifeTrigger" type="Trigger">
    <script>
      powerSensor = GetSensorSnapshot("PowerChargingSensor");
      if(powerSensor.StateExited = "Charging"){
        GetAction("BatteryLifeSurvey").Run();
      }
    </script>
  </trigger>
</triggers>
At home
Battery was getting low
Recharged using AC adapter

At Work
Needed to sync data
Recharged using USB
Further exploration could uncover:

• The average distance from home or work when suffering battery loss

• The primary reason people run out of battery (e.g., talk time, WiFi utilization)

• The number of places people charge their devices and the power source used.
Motivation

- 1 trillion SMS messages sent worldwide in 2005
- Explosive growth begs research:
  - Why SMS vs. voice?
  - Where do people use SMS?

Study

- Similar setup as before
- Asked questions after SMS sent
  - User’s location
  - Reason for using SMS
sms usage, mobility and response

**Location:** work  
**Reason:** couldn’t use voice

**Location:** home  
**Reason:** did not need immediate response

![Graph showing GSM Signal Strength (%)](image)

- **Estimated mobile periods**: 9 PM to 10 PM

---

23
Further exploration could uncover

- The link between mobility patterns and application usage
  - Do people SMS more when stationary than moving?
- How often users suffer from low cell signal strength and how this affects voice vs. sms
Initial 3-week study planned followed by longitudinal 3-month study

- Female participants from Seattle area
- Participants use *lab-provided* WM5 devices with ubifit instead of their own personal phones

UbiFit application

- Built off of MyExperience
- Collects both inferred activity and self-report activity data
- Data is sync’ed with Intel Research’s web server once/hr throughout the study
msp + my experience

![raw data](image1)

![inference data](image2)

Bluetooth

activities:
- standing
- sitting
- walking
- biking
- jogging

50%
subset of ubifit triggers

- **Journal reminder**
  - If journal has not been used in ~2 days and it’s past 8PM, launch journal reminder

- **Uncertain activity occurred**
  - If the system *knows* an activity occurred but couldn’t determine the exact activity, a survey is launched

- **MSP troubleshooter**
  - If the MSP hasn’t been seen in ~2 hrs and it’s after 10AM, launch a troubleshooter
Mobile therapy
   • Margie Morris, Bill Deleeuw, et al.
   • Digital Health Group, Intel

Multiple sclerosis pain and fatigue study
   • Dagmar Amtmann, Mark Harniss, Kurt Johnson, et al.
   • Rehabilitative Medicine, University of Washington

Smartphones for efficient healthcare delivery
   • Mahad Ibrahim, Ben Bellows, Melissa Ho, Sonesh Surana et al.
   • Various departments, University of California, Berkeley
1. **Coverage:** collect rich information about features of interest
   - Combines sensors traces + self-report

2. **Scale:** collect large amounts of data over long periods of time
   - Sensor traces scale well, context-triggered self-report can be used intermittently

3. **Extensible:** easily add new data collectors
   - Sensors, actions, triggers, user interface can be extended

4. **Situated:** collect *real* usage data in its natural setting
   - Runs on a user’s personal device

5. **Robustness:** protect or backup data collected in the field
   - Data can be opportunistically sync’d to research servers
thankyou

Source code available:
http://www.sourceforge.net/projects/myexperience

MyExperience is a context-aware data collection platform for capturing objective and subjective data as it's experienced.

jonfroehlich@gmail.com

Mike Chen, Sunny Consolvo, Beverly Harrison, and James Landay
3. What type of activity did you do?
1. Cardio
2. Walking
3. Strength Training
4. Flexibility Training
5. Other

4. What type of cardio activity did you do?
1. Running
2. Cycling
3. Elliptical trainer
4. Stair climber
5. Cardio class
6. Swimming
7. Roller-blading / Skating
8. Hiking
9. Rowing
platform support

- Currently Windows Mobile
  - SmartPhones and Pocket PCs
- 2006 Marketshare (Canalys Report 2006)
  - Symbian: 67%
  - Windows Mobile: 14%
  - RIM: 7%
  - Linux: 6%
- 2010 Estimates (The Diffusion Group 2006)
  - Microsoft will overtake Symbian for marketshare
- We are exploring a Symbian port
prelim researcher feedback

- **Surveyed 5 researchers**
  - All but one were experienced programmers

- **Positive comments**
  - The ability to “trigger anything in response to such a wide range of events or combination of events.”
  - “an easy way for a semi-technical designer to set up user-experience studies for cell phone applications”
  - “the XML structure is excellent and is deeply expandable through C# extensions to MyExperience”

- **Concerns**
  - Needed examples to understand how to use
  - Desire to have script debugging tools