Going Native
Relying on Pidgins and Creoles to Construct High Confidence Software

Travis Breaux
High Confidence Software and Systems Conference
May 6, 2014
Going Native

The ethnographer aims “to grasp the native's point of view, his relation to life, to realize his vision of his world”

—Bronislaw Malinowski, *Argonauts of the Western Pacific*, 1922
Boundary Objects and Trade Zones

- Patient diagnosis
- Radiation prescription
- Fields
- Source-to-skin distance
- Beam attenuation

- Computation for treatment dose
- Digitizing tablet for drawing shielding blocks
- Mechanical and digital readout alignment of gantry rotation to within 1°
- Digital timer is within 1 sec of stopwatch for irradiation times < 1-5 minutes
Failures in Critical Systems

High profile bugs yield critical failures

• Therac 25 (race condition)
• Ariane 5 (64- to 16-bit conversion)
• NASA Mars Climate Orbiter (metric v. non-metric)
• USS Yorktown (divide by zero)

Valid designs invalidated by the environment

• Theratron 780
• RQ-170 Sentinel
• USS Vincennes’ AEGIS Missile System
Overview of today’s keynote talk

- Abstractions must be grounded in the domain
  - Requirements and specifications
  - Example of a grounded theory

- System boundaries are constructed realities
  - Shared and unshared actions
  - Context of operation
  - Example composing specifications

- Assurance is an accumulation of evidence and shared understanding
  - Hazard analysis and refutation
  - Example in risk analysis
What are requirements?

• Scenarios and Use Cases
  "After the customer submits the purchase information and the payment has been received, the order is fulfilled and shipped to the customer’s shipping address”
  
• Optative statements
  "The system shall notify clients about their order status”
  
• Domain Properties and Assumptions
  "Every product has a unique product code”
  "Payments will be received after authorization”
**Semiotics (Ferdinand de Saussure, 1901)**

- **Signifier** is the form taken by a sign: an image, word or sound
- **Signified** is the concept represented by the sign

Saussure states two fundamental principles:

1. Signs are arbitrary
2. Signs are situated in space and time
What are specifications?

- Specifications are logical expressions of shared actions at the interface of the machine.
- This includes linking domain properties and agent actions as pre- and post-conditions.

\[ \forall s \ \forall c \ (\text{enrolled}(s, c) \Rightarrow \text{student}(s) \land \text{course}(c)) \]

• **Able**: Two important basic types are *student* and *course*. There is also a binary relation *enrolled*.

• Able defines these elements as follows:

\[ \forall s \forall c \ (\text{enrolled}(s, c) \Rightarrow \text{student}(s) \land \text{course}(c)) \]

• **Baker**: Do only students enroll in courses? I don’t think that’s true.

• **Able**: But that’s what I mean by student!
Designations as explanations

• If person is enrolled in a course, then the person is a student:
  \[ \forall s \forall c (\text{enrolled}(s, c) \implies \text{student}(s) \land \text{course}(c)) \]

• A person is a student, if and only if, there is a course where the student is enrolled
  \[ \forall s (\text{student}(s) \iff \exists c \text{enrolled}(s, c)) \]
Four Kinds of Denial

- **Denial by prior knowledge** – we have done this before, so we know what is required
- **Denial by hacking** – our fascination with machines dominates our focus on the how
- **Denial by abstraction** – we pursue elegant models which obscure, remove or downplay the real world
- **Denial by vagueness** – imply (vaguely) that machine descriptions are actually those of the world

Discovering grounded theory

- Grounded theory is systematically derived from a data set (Corbin & Strauss, 2008)
  - Derived from stylized system descriptions by multiple analysts
  - Inferences are only valid for that data set
  - Repeated studies needed to generalize results

- Statistics (Kappa/Alpha) to compute agreement above chance
  - If coding frame meets certain statistical assumptions

Grounded theory and privacy

Maximize Data Utility
• Collect everything, value is realized later
• Ensure open access; this drives innovation
• Disclose to leverage third-party value
• Retain as long as practical (longitudinal/behavioral)
• Avoid destruction

Minimize Privacy Risk
• Limit collection based on stated needs
• Limit access, obtain consent for new uses
• Limit disclosure and third-party uses
• Destroy when no longer needed
• Embrace destruction
Why privacy policies?

- U.S. and other laws require companies to have privacy policies
- In the U.S., companies should write honest privacy policies or face the wrath of regulators
- Ambiguity in policies allows for flexibility, which can be desirable in requirements
- Policies change to match evolving business practices (Google, Facebook, etc.)
Example in grounded theory
Begin with relevant questions

Privacy Principles (HEW Report, 1973)

• Are purposes stated for all collections (purpose specification)?
• Is data being over-collected (collection limitation)?
• Is data being re-purposed (use limitation)?

Desiderata

• Are there any conflicts in this policy?
• Are there nocuous ambiguities?

Mapping policy statements to types

- **Policy Statements** describe events or states outside the app
  
  “You must not violate any law or the rights of any individual or entity.”

- **Non-data Requirements** describe non-data functionalities
  
  “You will include your privacy policy URL in the App Dashboard.”

- **Data Requirements (~50%)** describe actions on data
  
  “You must not include functionality that proxies, requests or collects Facebook usernames or passwords.”
Step 1-2: Manually annotate policy text

Modal phrase “will” indicates an assumed permission
Transfer keyword
Datum
Target
Purpose
We will provide your information to third party companies to perform services on our behalf, including payment processing, data analysis, e-mail delivery, hosting services, customer service and to assist us in our marketing efforts.

Step 3: Write expression in specification language

SPEC HEADER
P performing-services > data-analysis, e-mail-delivery, hosting-services, customer-service, assisting-with-marketing-efforts

SPEC POLICY
P TRANSFER information TO third-party-company FOR performing-services
**Phrase heuristics used in mapping**

Action keywords indicate when a statement was coded as a collection, use or transfer requirement

<table>
<thead>
<tr>
<th>DL Action</th>
<th>Action keywords</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLLECT</td>
<td>Access, assign, collect, collected, collection, collects, give you, import, keep, observes, provide, receive, record, request, share, use</td>
</tr>
<tr>
<td>USE</td>
<td>Access, accessed, communicate, delivering, include, matches, send, use, used, uses, using, utilized</td>
</tr>
<tr>
<td>TRANSFER</td>
<td>Access, disclose, disclosed, disclosure, give, in partnership with, include, make public, on behalf of, provide, see, share, shared, transfer, use, used with, utilized by</td>
</tr>
</tbody>
</table>
Using context in annotation

• [Zynga] “may access and store some or all of the following information, as allowed by you, the SNS and your preferences”
  Action is COLLECT

• [AOL] “Personal information such as name, address and phone number is never accessed for this purpose”
  Action is USE

• [AOL] “In that the case, the acquiring (or merging) company will have access to your information”
  Action is TRANSFER
Step 3: Write expression in specification language

SPEC HEADER
  - $P \text{ performing-services} > \text{data-analysis, e-mail-delivery, hosting-services, customer-service, assisting-with-marketing-efforts}$
  - $A \text{ third-party } = \text{third-party-company}$
  - $D \text{ personally-identifiable-information} \not\subset \text{ip-address}$

SPEC POLICY
  - $P \text{ TRANSFER information TO third-party-companies FOR performing-services}$

Step 4: Compile language into Description Logic

$\text{payment-processing} \sqsubseteq \text{performing-services}$
$\text{e-mail-delivery} \sqsubseteq \text{performing-services}$

\ldots

$p_2 \equiv \text{TRANSFER} \sqcap \exists \text{hasObject.information} \sqcap \exists \text{hasTarget.third-party-company} \sqcap \exists \text{hasPurpose.performing-services}$

$p_2 \sqsubseteq \text{Permission}$
Design of the language

• Expressing Modality in DL
  • Obligation $\sqsubseteq$ Permission
  • $Conflict \equiv Permission \sqcap Prohibition$

• Actions
  • Collect, Use and Transfer

• Actions have following DL Roles
  • hasObject.Datum – the object of the action (data element)
  • hasSource.Actor – the source of the object (an actor)
  • hasPurpose.Purpose – the purpose of the action
  • hasTarget.Actor – the recipient of the object (an actor)

Do “they” transfer your PII to advertisers?

- We share your personal information with advertisers
- We share information for marketing
  
  information > personal-information

- We share information with third parties
  
  advertisers < third-parties

- We do not share information, unless you consent
  
  TRANSFER information TO anyone FOR anything
How we identify conflicts

p1: Permitted to transfer IP address to anyone for advertising

r2: Prohibited from transferring IP address to third-party advertisers for anything

Permitted Transfer = p1

Zynga
Third-party advertiser

Delivering advertisement
Payment processing
Marketing third-party
Target advertising

r2 = Prohibited Transfer

Actor
Datum
Purpose
Grounded Theory, Summary

- Multiple coders establish statistical inter-rater reliability
- Domain experts inspect translation (structure) and inferences drawn (reasoning) – models are boundary artifacts
- Evaluation of the qualitatively-derived theory [Yin, 2013]
  - Construct Validity – are we measuring what think we are?
  - External Validity – how well does the model generalize?
  - Internal Validity – are all conclusions valid?
  - Reliability – can others create the same models?

Parnas’ Four Variable Model

- **Environment**
  - monitored variables

- **Input Devices**
  - input data

- **Software**
  - output data

- **Output Devices**
  - controlled variables

- **Environment**
  - OUT
  - (properties of the output devices)

- **Input Devices**
  - IN
  - (properties of the input devices)

- **Software**
  - SOFT
  - (properties of the software)

**REQ** (relationships between monitored and controlled variables that the system is required to establish and maintain)

**NAT** (natural relationships between monitored and controlled variables that are part of the domain)


©2014 T.D. Breaux
The driver shall decrease their vehicle speed when entering a school zone.

When a wheel loses traction, the car shall “pump” the break to reduce the wheel’s rotation velocity.

Shared and unshared actions

- Actions are environment- or machine-controlled
- Actions either:
  - Shared with (belongs to, is observable by) the machine
  - Unshared, and not observable by the machine

### Actions in a Turnstile

<table>
<thead>
<tr>
<th>Actions</th>
<th>Shared</th>
<th>Unshared</th>
</tr>
</thead>
<tbody>
<tr>
<td>pay, push</td>
<td>enter</td>
<td></td>
</tr>
</tbody>
</table>
USS Vincennes Incident

**P-3 Orion** presenting classic targeting profile

**USS Vincennes** engaged in high speed surface battle with Iranian gunboats

**Iranian Flight 655**
- Take off from joint military/civilian runway, 27 minutes behind known commercial schedules
- Within known commercial air corridor, off centerline by 3-4 miles
- Ambiguous mid-range altitude
- Perceived altitude as decreasing

Past experiences, including recent military incidents and tactics, affected decision making.
Carnegie Mellon University

What is Waze?
Example service integration

**Facebook Connect**
- Authentication Services
- Basic profile information, email address, age, list of friends, photo

**Waze**
- Driving route, location, speed, device ids

**Flurry.com**
- Mobile Analytics & Advertising Services

**Legend:**
- Data flow

Scalable Storage Services
Characterizing data flows in DL

- **Underflow**, occurs when the data source $F_s$ is subsumed by the target $F_t$, if and only if,
  $$T \models F_s,j \sqsubseteq F_t,k$$

- **Overflow**, occurs when the data target is subsumed by the source, if and only if,
  $$T \models F_t,j \sqsubseteq F_s,k$$

- **Exact flow**, occurs when the data source and target are equivalent, if and only if,
  $$T \models F_s,j \equiv F_t,k$$

- **No flow**, otherwise.
Tracing multi-party data flows

Waze

- W_P6
  - COLLECT personal-information FROM waze-user FOR enhancing-service-experience

- W_P46
  - TRANSFER unique-device-id TO ad-networks

Flurry

- F_P1
  - COLLECT device-id, device-os, mac-address

Assume: unique device id is part of personal information
Assume: unique device id is a synonym for device id and mac address

Example from Waze and Flurry.com privacy policy
Tracing multi-party data flows

Example from Waze and Flurry.com privacy policy

©2014 T.D. Breaux
“If you have received consent from your End Users, you may submit ad requests which may include device identifiers (including…), age ranges, gender, device type, OS, language, Personas, latitude and longitude (obfuscated by Flurry… for end users with Android devices), and customer key values for the purpose of selling your inventory through the Flurry RTB Marketplace. Such data may be used by media buyers to optimize their bids on impressions and focus their advertising purchases, and for retargeting ads to End Users.”

Excerpt from Flurry.com Privacy Policy, last updated July 19, 2013
Interlingua to align two policies

NS1 http://localhost/waze-pp.owl customer
NS2 http://localhost/flurry-pp.owl ad-networks

D ads-clicked < aggregated-data
D ads-clicked = clicks
D ads-posted < aggregated-data
D ads-viewed < aggregated-data
D age = age
D list-of-friends < end-user-data
D location = location
D personally-identifiable-information < end-user-data
D profile-picture < end-user-data
D unique-device-id = device-id
**Waze Collections & Transfers**

- **WP19**: COLLECT information* FROM social-network
- **WP20**: TRANSFER PII FROM social-network
- **WP26**: COLLECT location
- **WP54**: TRANSFER location TO ad-companies
- **WP46**: TRANSFER unique-device-id TO ad-networks
- **WP6**: COLLECT personal-information FROM waze-user FOR enhance-personal-experience

**Legend:**
- User’s social network information, including name, age, gender
- User’s mobile device location
- User’s mobile device unique identifier

**Flurry Collections & Transfers**

- **FP20**: COLLECT ad-requests FROM customer FOR sales-in-rtb-marketplace
- **FP20**: TRANSFER end-user-data TO advertising-partners
- **FP1**: COLLECT device-id
- **FP1**: TRANSFER device-id, location TO applicable-publisher
- **FP13**: TRANSFER end-user-data TO advertising-partners

©2014 T.D. Breaux
Software assurance is an accumulation of multiple forms of evidence

- Prototypes
- Simulations
- Models
- Benchmarks
- Test cases
- Audit logs
On the Ground – 1

- The Airbus A320-200 airplane has a software-based braking system that consists of:
  - Ground spoilers (wing plates extended to reduce lift)
  - Reverse thrusters
  - Wheel brakes on the main landing gear

- To engage the braking system, the wheels of the plane must be on the ground.
There are two “on ground” conditions:
1. Either shock absorber bears a load of 6300 kgs
2. Both wheels turn at 72 knots (83 mph) or faster

Ground spoilers activate for conditions 1 or 2

Reverse thrust activates for condition 1 on both main landing gears

Wheel brake activation depends upon the rotation gain and condition 2
Organizing Arguments

Appeal to the realities of specification
• Restricted data is removed from third party disclosures
  Source: Verified by proof

Appeal to policies and directives
• Remote access must use VPN with blocking mode enabled
  Source: DoD 8500.2, Control EBRP-1, Feb 6, 2003

Appeal to studies of human behavior
• 39% polled access sensitive data over public WiFi
  Source: 2014 Harris Poll
Trust Assumptions

“Entities in a relation trust each other to have or not have certain properties (so-called trust assumptions)” [Viega & McGraw, 2002]

Example Trust Assumptions:

- Users won’t expose passwords
- Users only connect from secure networks
- Only authorized users have access to the system

Toulmin Model of Argumentation

Claim

Data is accessed by authorized users

Mobile computers outside firewall

Rebuttal

Users connect from secure networks

Grounds

Backing

RSA tokens are issued to users

Warrant

Remote access is over secure VPN

Theory of Situation Awareness
Going beyond “usability” of methods and tools

**Perceive** cues in system artifacts

**Comprehend** meaning of the cues

**Project** consequences

**Decide** how to mitigate consequences

I don’t see an NTP server on this network, but I know that Windows Domain Controller can act as NTP, so I am going to assume that when they install it they’ll probably leave that box checked because it’s a default option. I think that is probably happening here.
Variations in Classic SA Patterns*

<table>
<thead>
<tr>
<th>Name</th>
<th>Pattern</th>
<th>Freq.</th>
<th>Accu.**</th>
<th>Participants</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classic w/o Decision</td>
<td>P→C→J</td>
<td>4</td>
<td>100%</td>
<td>P1, P3, P6</td>
</tr>
<tr>
<td>Projection-Decision</td>
<td>J→D</td>
<td>31</td>
<td>90%</td>
<td>All except P1</td>
</tr>
<tr>
<td>Skip Projection</td>
<td>P→C→D</td>
<td>10</td>
<td>100%</td>
<td>P1, P3, P4, P6, P11</td>
</tr>
<tr>
<td>Skip Perception</td>
<td>C→J</td>
<td>55</td>
<td>81%</td>
<td>All</td>
</tr>
<tr>
<td>Skip Perception-Projection</td>
<td>C→D</td>
<td>56</td>
<td>83%</td>
<td>All except P2 &amp; P5</td>
</tr>
<tr>
<td>Perception-Comprehension</td>
<td>P→C</td>
<td>61</td>
<td>81%</td>
<td>All except P10</td>
</tr>
</tbody>
</table>

**Legend:** P-Perception, C-Comprehension, J-Projection, D-Decision

* Results based on 2,595 coded statements from ~5.3 hours of interviews with 11 participants; Cohen’s Kappa = 0.44-0.77

**Accuracy is the proportion of pattern instances that match pattern definition
Traceability and Evolving Assurance

P → Q ∨ R

P

∴ Q ∨ R

Domain Artifacts & Transcripts

Engineer

inspected, evaluated

Formal Models & Abstractions

Trust Assumptions & Assurance Cases

Architecture

Correct Code

(Re-)Certification

Domain Natives
Talk Summary

- Grounded analysis to discover the “right” semantics for expressing the problem space

- System boundaries delineated, and the consequent shortcomings thereof understood and accepted

- Evaluate formal methods using meta-cognitive models to iterate over human conceptions of the solution space
  - How do engineers and domain experts use specification to analyze systems?