Carnegie Mellon

isr institute for SOFTWARE RESEARCH

Detecting Repurposing and Over-collection in Multi-Party Privacy Requirements Specifications

Travis D. Breaux, Daniel Smullen, Hanan Hibshi

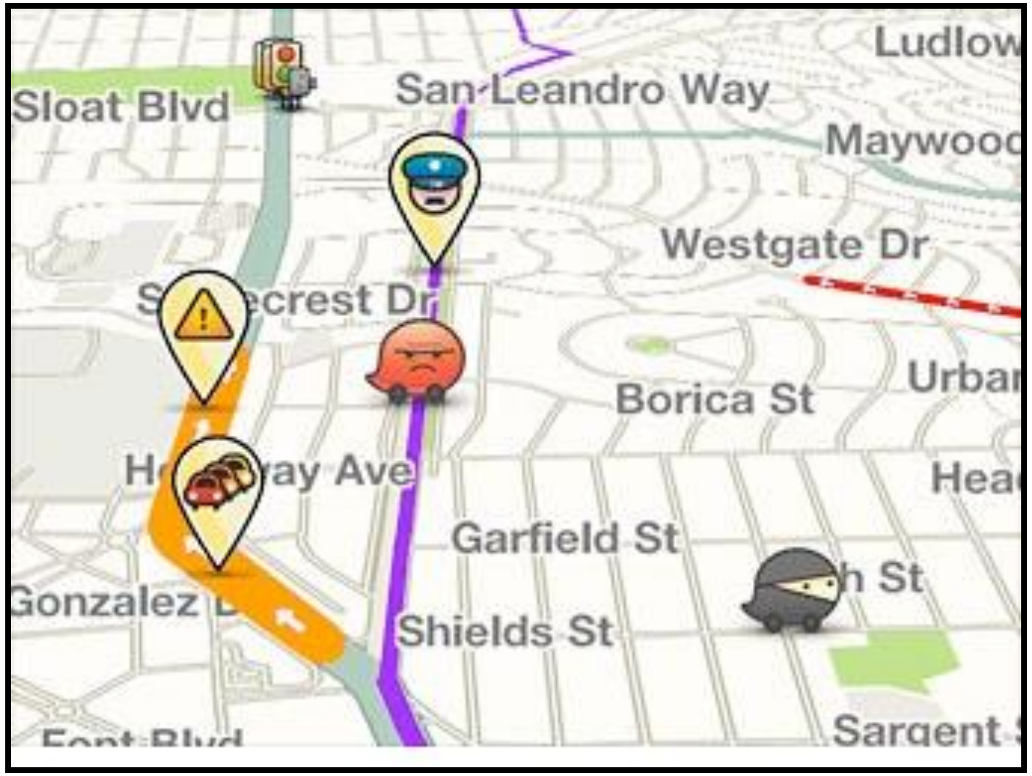
{breaux,dsmullen,hhibshi}@cs.cmu.edu

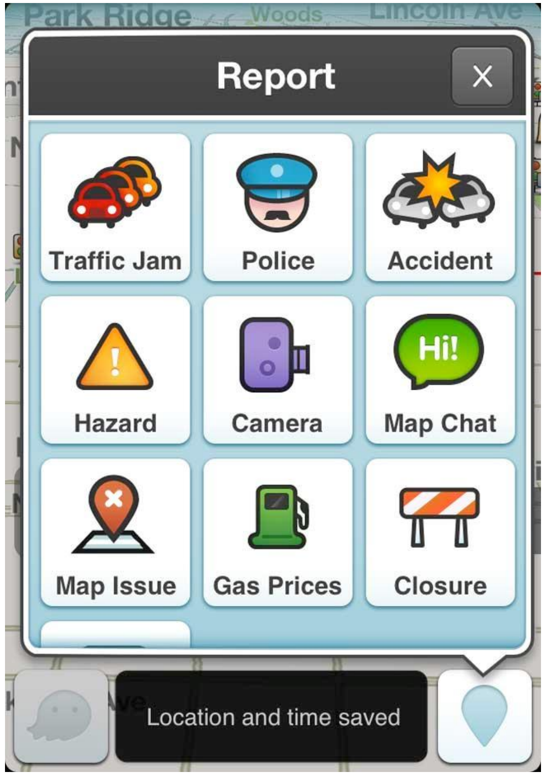
Motivation

- Mobile and web applications are using Service Oriented Architectures more and more.
- How do we ensure that using 3rd party services doesn't increase our privacy risk?
  - What sort of data do they want?
  - What will they do with my data once they have it?
  - What am I willing to give them?
- Tough to answer these questions.

What is waze ?

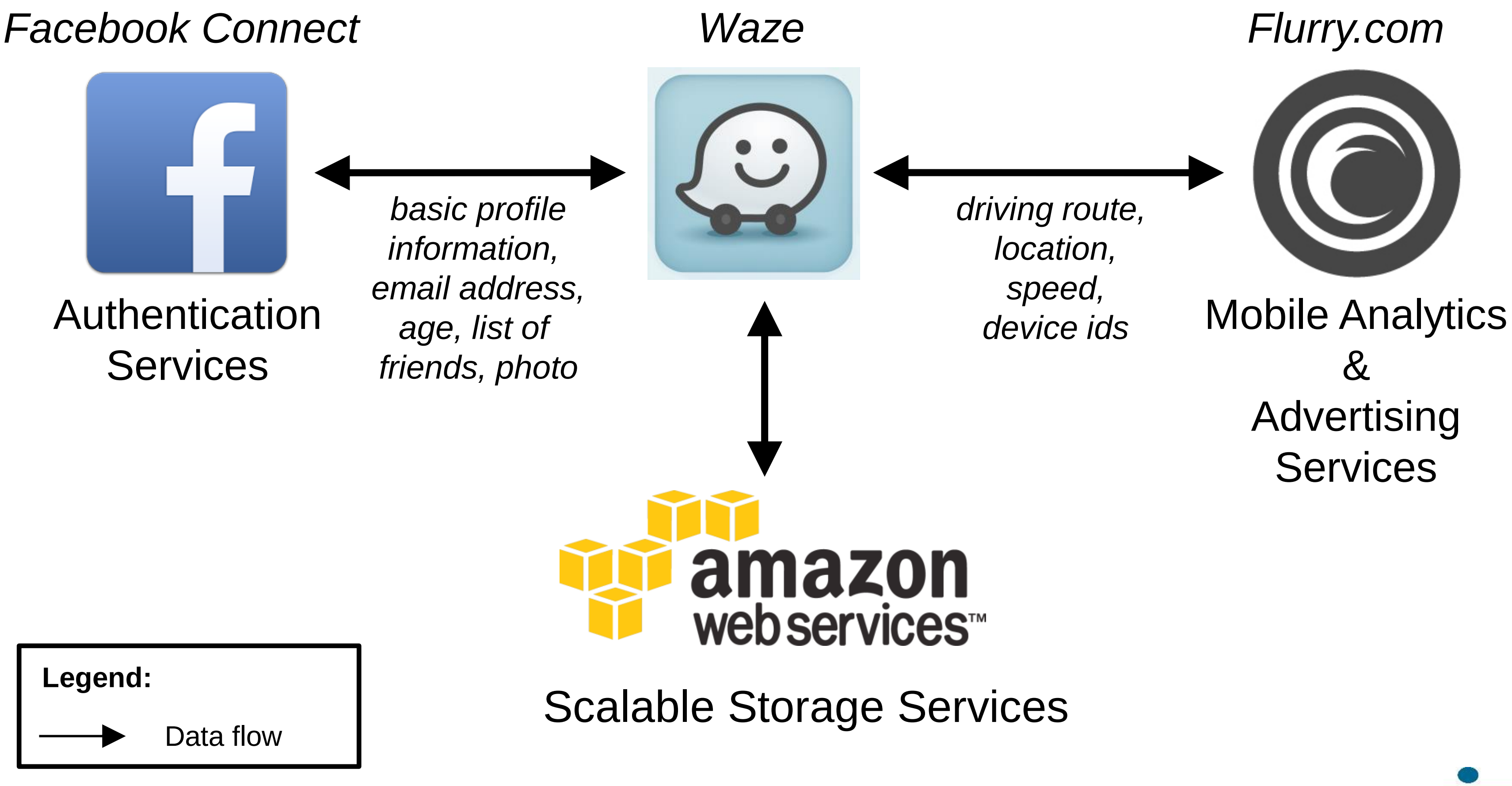
OUTSMARTING TRAFFIC, TOGETHER





Location and time saved

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

Scalable Storage Services

Legend:

Data flow

Case Study Research Questions

RQ1: What conflicts exist in our formalization of Waze’s policies?

RQ2: What multi-party data flows exist?

RQ3: Does data repurposing or over collection occur?

Building on Previous Work

- [BR13] introduced Eddy.
- SQL-like syntax for policy specifications.
- Limited to tracing policies within a system; can't extend to 3rd parties.
- Great for finding conflicts in policies (conflicting interpretations).
- Some performance analysis.

[BR13] T.D. Breaux, A. Rao. “Formal Analysis of Privacy Requirements Specifications for Multi-Tier Applications, *21st IEEE International Requirements Engineering Conference*, pp. 14-23, Jul. 2013

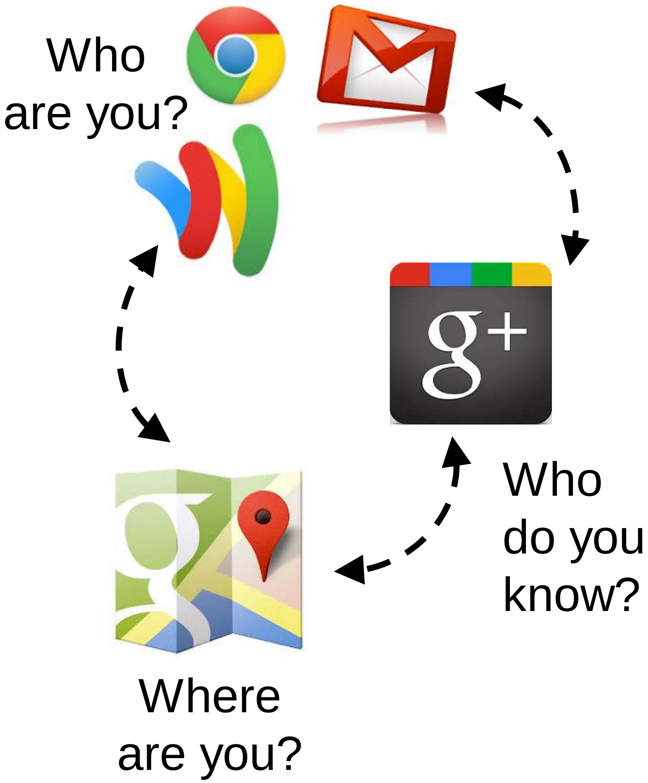
Related Work

- Extracting goals from privacy policies
  [Antón et al.,2004; Breaux & Antón, 2005; Young et al., 2011]
- Formal models of privacy-related requirements
  [Breaux, Hibshi, Rao, 2013; Liu et al. 2003; Tun et al. 2012; Omoronyia et al., 2013]
- Static and dynamic analysis of code (TaintDroid, Appfence, Pscout)
  [Enck et al., 2010; Hornyack et al. 2011; Yee Au et al. 2012]
- Multiple policy-related languages…

The Value of Knowing

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



Who are you?

Who do you know?

Where are you?

Balancing Utility and Risk

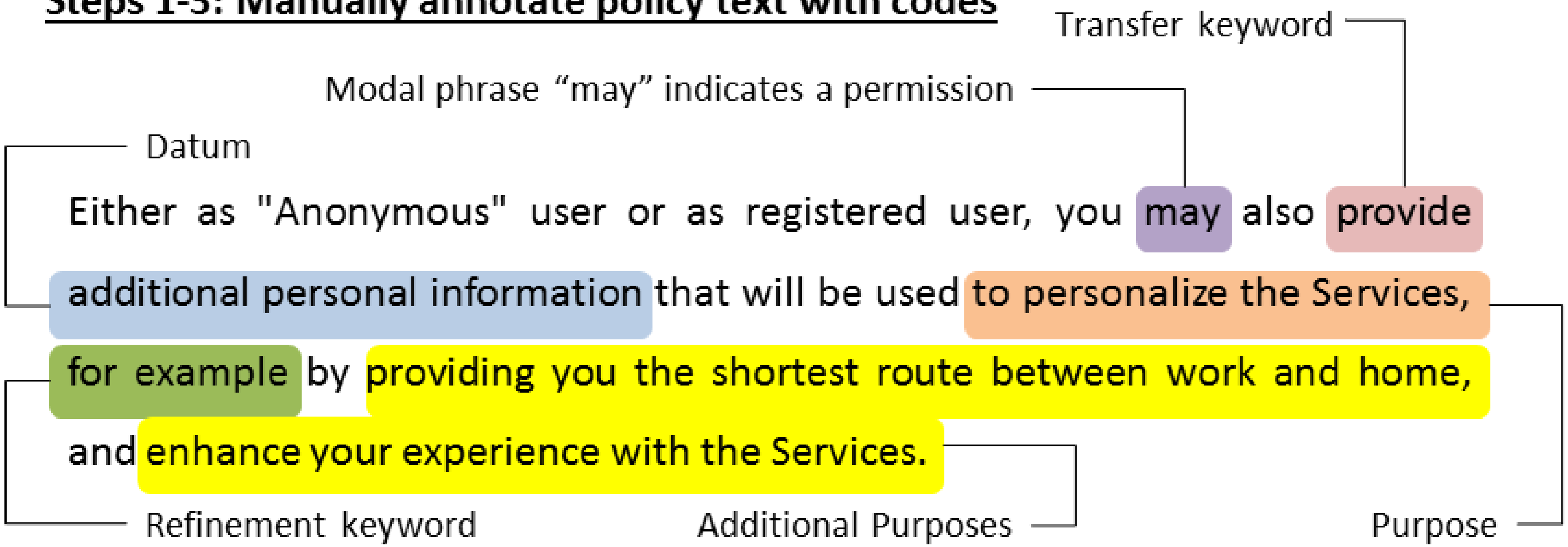
Maximize Data Utility

- 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

Steps 1-3: Manually annotate policy text with codes



Transfer keyword

Modal phrase "may" indicates a permission

Datum

Either as "Anonymous" user or as registered user, you may also provide additional personal information that will be used to personalize the Services, for example by providing you the shortest route between work and home, and enhance your experience with the Services.

Refinement keyword

Additional Purposes

Purpose

Step 4: Write expression in Eddy (re-topicalized for Waze)

```
SPEC-HEADER
    P personalizing-services > providing-shortest-route, enhancing-service-experience
SPEC-POLICY
   P COLLECT personal-information FROM waze-user FOR personalizing-services
```

Step 5: Tool compiles Eddy into Description Logic

(A) $\text{providing-shortest-route} \sqsubseteq \text{personalizing-services}$

(B) $\text{enhancing-service-experience} \sqsubseteq \text{personalizing-services}$

(C) $p_6 \equiv \text{COLLECT} \sqcap \exists \text{hasObject.personal-information} \sqcap$
$\exists \text{hasSource.waze-user} \sqcap \exists \text{hasPurpose.personalizing-services}$

(D) $p_6 \sqsubseteq \text{Permission}$

Three Privacy Principles

- *Purpose specification principle:*
  - The purposes for which data is collected should be explicitly stated.
- *Collection limitation principle*:
  - Collection of personal data should be limited (to what will be used).
- *Use limitation principle:*
  - Uses should be limited to the purposes for which the data was originally collected, and nothing else.

- Exceptions for consent and legal compliance.

Three Privacy Principles

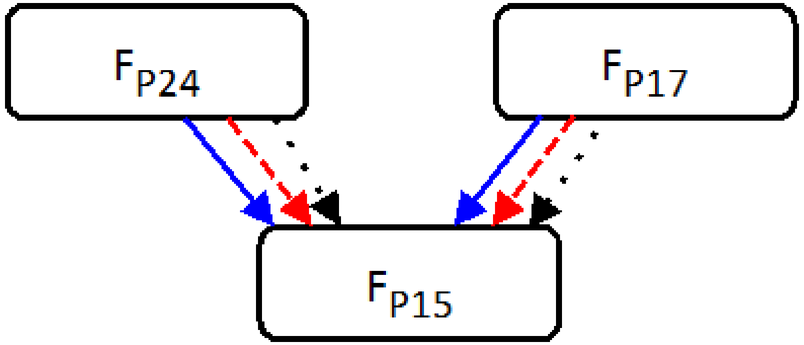
- Commonly accepted.
  - U.S. Fair Information Practice Principles (FIPPs)
  - OECD Guidelines on the Protection of Privacy and Trans-border Flows of Personal Data

- If these principles are violated, there are privacy risks.
  - Repurposing
  - Overcollection

Crossflow Analysis (1st Party), Identifying Data Over- and Under-flows

P COLLECT device-id, ip-address, FROM end-user FOR anything

P.COLLECT device-id, location, … FROM application FOR anything



P TRANSFER device-id FROM anyone FOR anything

Legend:

← hasObject
⇠ hasSource
⇠ hasPurpose
Blue: overflow
Red: underflow
Black: exact flow

Example from Flurry.com privacy policy, last updated July 19, 2013

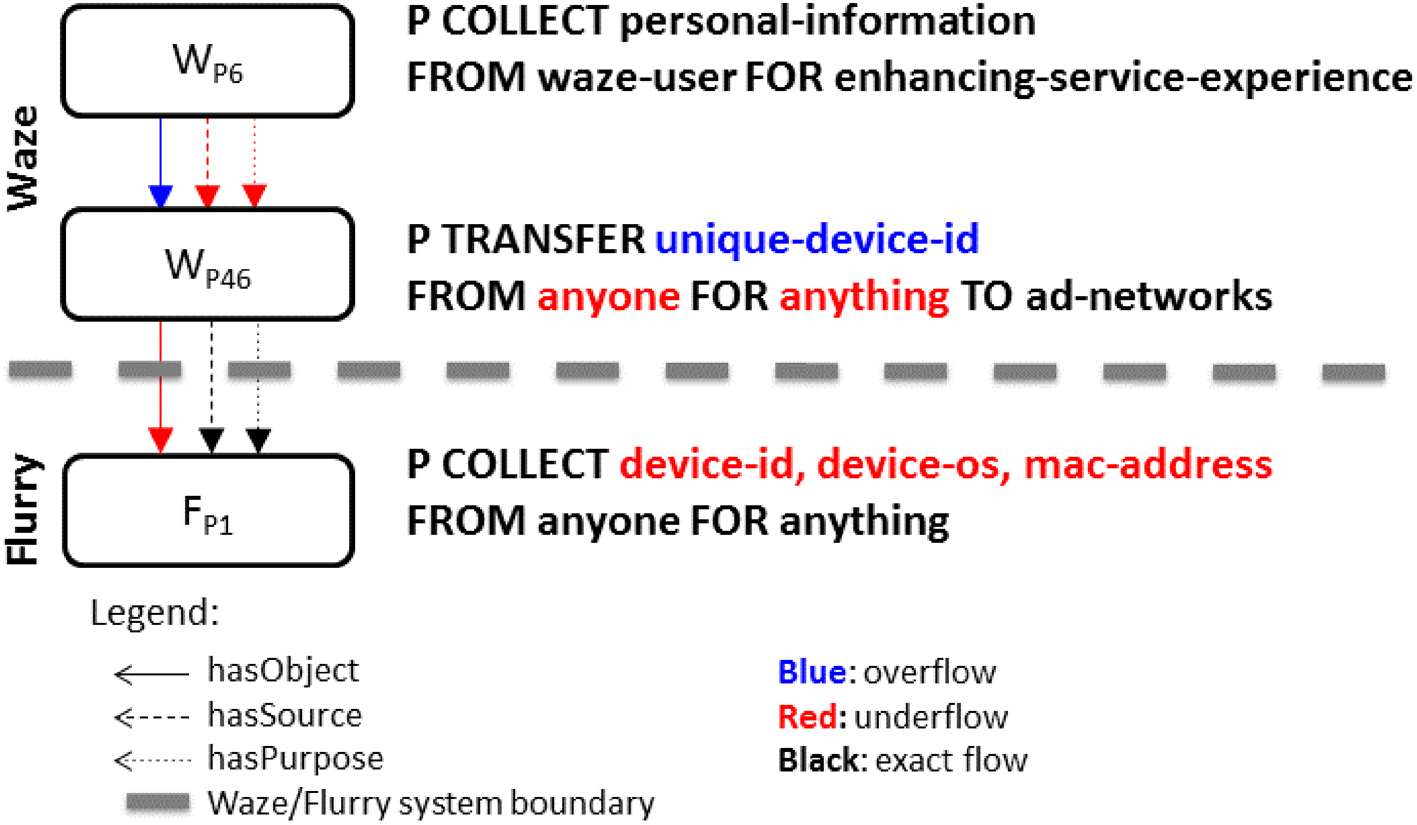
Tracing to 3rd Parties

- Requires a dictionary, to map each party’s lexicon.

- Your definition of information is different to mine.
- Your definition of a purpose is different to mine.
- And so on…

- Dictionaries can be developed separately by different parties.

Crossflow Analysis (3rd Party)



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

P TRANSFER unique-device-id FROM anyone FOR anything TO ad-networks

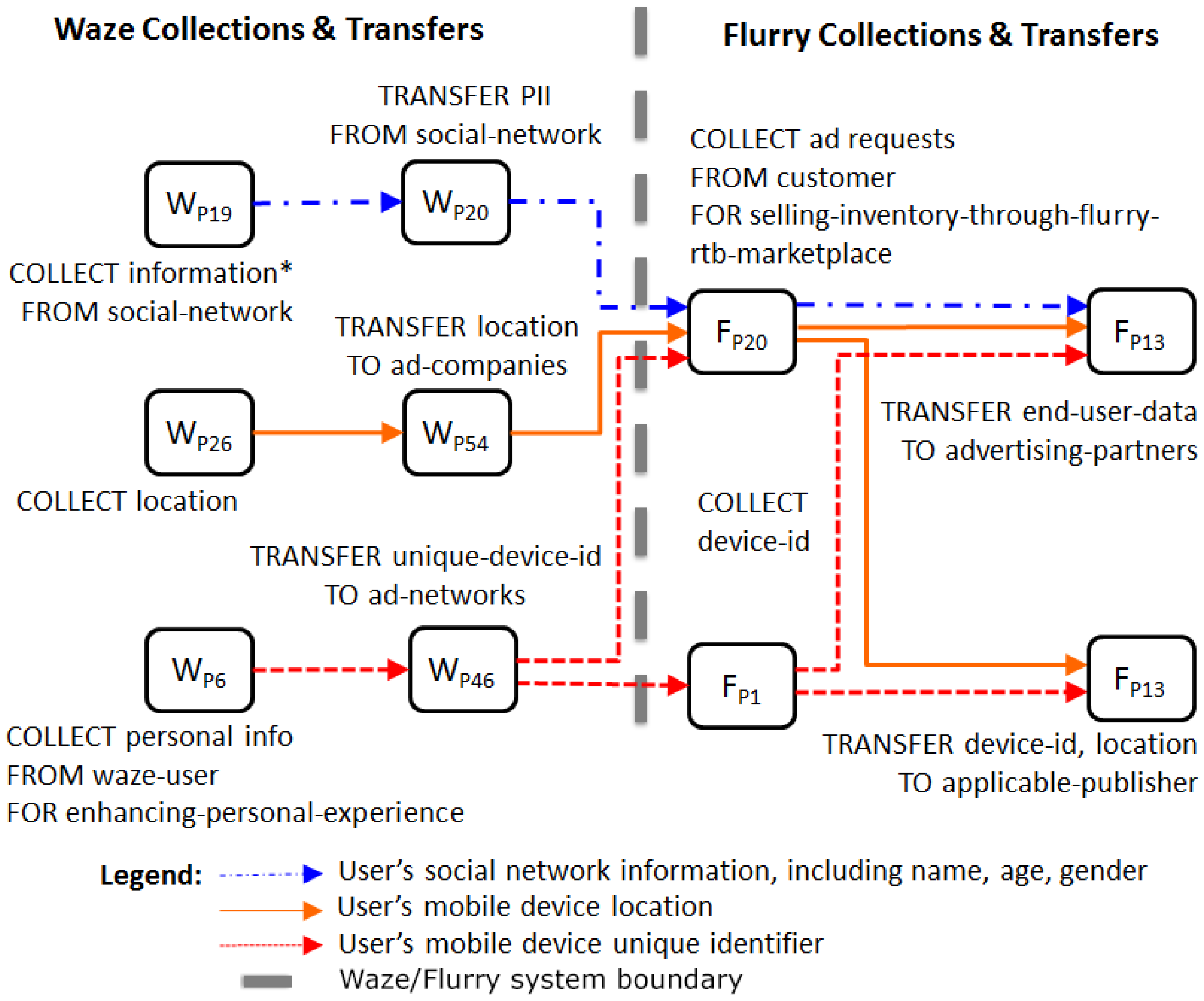
P COLLECT device-id, device-os, mac-address FROM anyone FOR anything

Legend:

← hasObject
<- - - - hasSource
<········ hasPurpose
▬ Waze/Flurry system boundary

Blue: overflow
Red: underflow
Black: exact flow

Crossflow Analysis (3rd Party)



Waze Case Study Results

Policy	Total Stmts	Data Req’ts	Modality[1]			Actions[2]		
			P	O	R	C	U	T
Waze	150	65	60	0	5	13	18	34
Flurry	155	44	42	0	2	15	6	23
Facebook	136	55	24	1	30	13	24	18

Overview of Requirements

1. Privacy policies generally describe permissions (P), with few prohibitions (R) and almost no obligations (O)
2. Data requirements describe only collect (C), use (U) and transfer (T) actions, which comprised 28-43% of total policy

Patterns: (Purpose Hoisting, Unrestricted Cross-Flows)

Waze Case Study Results

Policy	Definitions		Axioms			Concepts		
	Expl.	Impl.	S	D	E	A	D	P
Waze	19	29	41	3	4	6	29	13
Flurry	14	20	21	1	12	0	34	0
Facebook	13	0	11	0	2	0	13	0

Ontology Complexity

- Inferences to discover implied (Impl.) definitions (e.g., *personal information* is equivalent to *personal details*).
- Formalisms: Subsumption (S), Disjointness (D) and Equivalence axioms (E).
- Concepts: Actors (A), Data types (D) and Purposes (P)

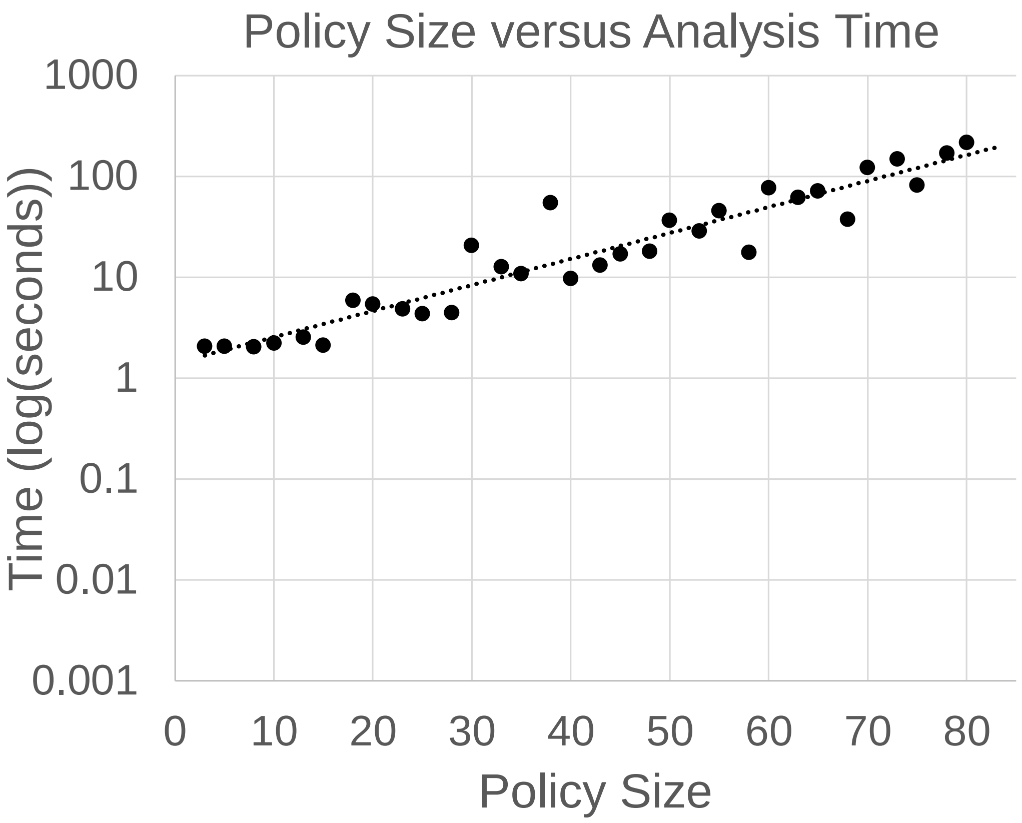
Waze Case Study Results

- Found conflicts – envision tool-driven techniques to help specification authors to detect and remove these.
- Discovered attacks on collection and use limitation principles (unrestricted crossflows, purpose hoisting)
- Scaling tools to larger policies (akin to map-reduce).
- Expand to other data practices: consent, data retention, etc.

Quick note on performance analysis…

- Most policies have around 50 data requirements.
- Gets bigger when you introduce multiple parties.
- So, how big can we make them before the Eddy toolchain blows up?
    - AKA Does it scale?

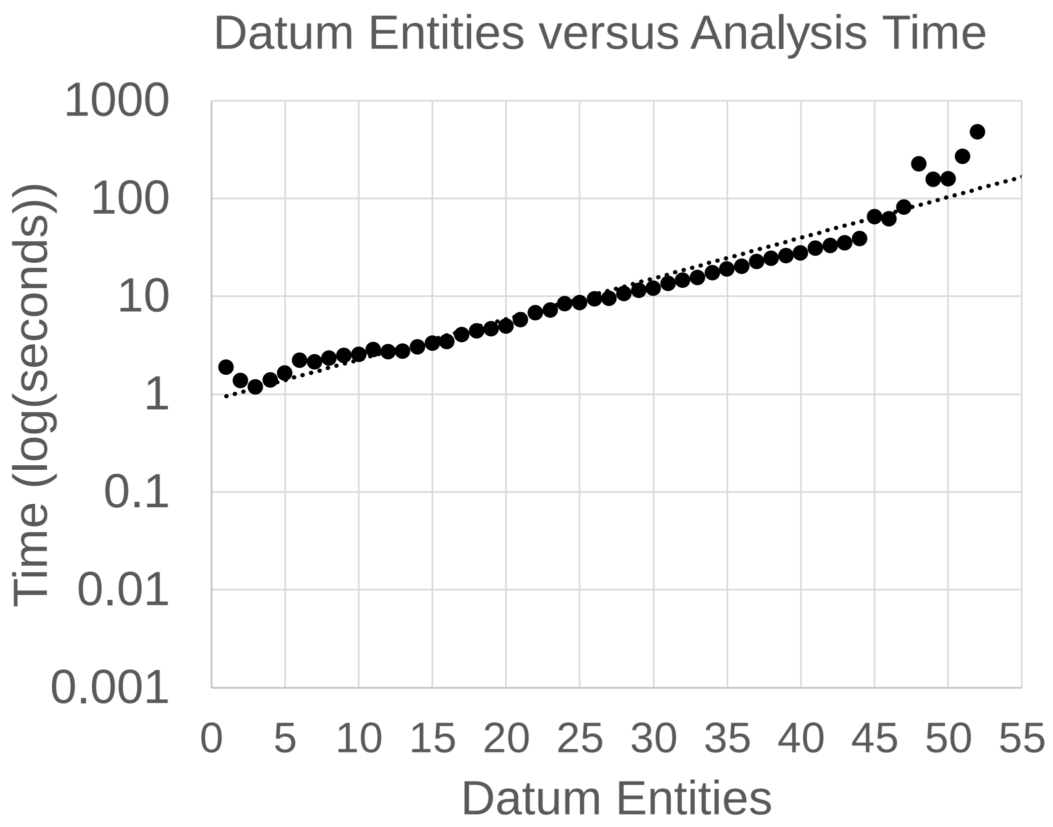
Does it Scale?



- Logarithmic plot.
- How long does it take to do analysis as the number of requirements grows?

- 80: Under 4 minutes.

Does it Scale?



- New benchmark.
- Logarithmic plot.
- How long does it take to do analysis as the number of data types grow?
- Policy size: fixed, 400.
- 52: Under 8 minutes.

Conclusions

- Eddy works equally well with multi-party compositions.
- Toolchain scales well to extremely large policies.
- Using two coders and the toolchain, we can analyze a complex compositional system.
- Validate conformance to the 3 privacy principles.
- Two interesting privacy design patterns were found.