Preventing Data Errors with Continuous Testing

Kıvanç Muşlu Yuriy Brun Alexandra Meliou

University of Washington University of Massachusetts

Data entry error wipes out

articles.chicagotribune.com/2010-12-02/business/ct-biz-1202-problem-mcshane-20101202_1_prudential-em...

Alexandra

Chicago Tribune

HOME NEWS BUSINESS SPORTS A&E LIFESTYLES OPINION REAL ESTATE CARS JOBS

Data entry error wipes out life insurance coverage

Related Articles

Oakland Unified makes $7

archive.oaklandlocal.com/article/oakland-unified-makes-76-million-accounting-error-budget-asking-schools-...

Alexandra

oakland LOCAL

This is the Oakland

Oakland Local i

ABOUT STORY CATEGORIES

Oakland Unified makes $7

schools not to count on it

Published on Thursday, February 07, 201

Last updated on 05:30PM, Monday, Febru

VERNON HAL Deputy Superintendent, Business & Operat

February 9, 2013

Recently, we informed you of a data entry error that inflated t

all schools sites by a cumulative total of $7.6 million. We apo

also want to clarify what happened, the impact it had, and ou

Prior to Winter Break, site administrators received a one-pag

total allocations for each site. **The summary indicated that**

allocations, was flat, essentially the same as the current

allocations were loaded in the tool used to plan expenses. T

present the full picture of expenses incurred by the site – it a

the site level.

In this instance, an amount equal to the utility costs for a sch

the tool associated with central office costs, and a second tim

The latter allocation was a mistake and made the site budget

Data entry is a top cause o

www.amednews.com/article/20050124/profession/301249959/4/

Alexandra

amednews.com

AMERICAN MEDICAL NEWS

HOME | PAST WEEKS | ARCHIVES | TOPICS | COLUMNS | MULTIMEDIA Published by the American Medical Association

SECTIONS » Government | Profession | Business | Opinion | Health Search Articles Search tips

TOPICS »
Health Reform
EHRs
Medicare
Liability
AMA House
» More

COLUMNS »
Contract Language
Ethics Forum
In the Courts
Practice Management
Technically Speaking
» More

LISTINGS
Issue dates
Regions
Columns
Archives
Writers

PROFESSION

Data entry is a top cause of medication errors

■ Training and design are seen as keys to reducing electronic prescribing errors.

By ANDIS ROBEZNIEKS — Posted Jan. 24, 2005

PRINT | EMAIL | RESPOND | REPRINTS | LIKE | SHARE | TWEET

Computerized prescribing systems might cut the quantity and severity of medication mistakes, but they can't eliminate them entirely, said patient safety experts who reviewed the U.S. Pharmacopeia's 5th annual study of medication error reports.

WITH THIS STORY:
» The chief reasons
» Most errors don't cause harm
» External links
» Related content

The study of the more than 235,000 error reports submitted in 2003 by 570 health care facilities was the largest ever by USP. And as the number of reported errors goes up, the percentage that causes patient harm has gone down. But the findings that generated the most discussion are those indicating that electronic prescribing is creating new types of errors.

"Computer entry" was the fourth-leading cause of errors, accounting for 13% (27,711) of the medication errors reported in 2003. In contrast, illegible or unclear handwriting was the 15th-leading cause, and accounted for 2.9% (6,134) of reported errors.

ADVERTISEMENT
ADVERTISE HERE

FEATURED

Confronting bias against obese patients

■ Medical educators are starting to raise awareness about how weight-related stigma can impair patient-physician communication and the treatment of obesity. **Read story**

Goodbye

■ *American Medical News* is ceasing publication after 55 years of serving physicians by keeping them informed of their rapidly changing profession. **Read story**

Policing medical practice employees after work

■ Doctors can try to regulate staff actions outside the office, but they must watch what

Waiting for m.trb.co

https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CB8QFjAA&url=http%3A%2F%2Fwww.amednews.com%2Farticle%2F20050124%2Fprofession%2F301249959%2F4%...

Data discrepancies

FlightView

American Airlines Flight Number 119 (AA119)

FLIGHT TRACKER

Departure

Airport:
Scheduled Time: 6:15 PM, Dec 08
Takeoff Time: 6:53 PM, Dec 08
Terminal - Gate: Terminal A - 32

ArrivalStatus: In Air

Airport:
Scheduled Time: 9:40 PM, Dec 08
9:42 PM, Dec 08
Estimated Time:
Track This Flight ve!
Time Remaining: 25 min
Terminal - Gate: Terminal 4 - 42
Baggage Claim: 4

6:15 PM

9:40 PM

FlightAware

AAL119 (Track inbound flight)
(web site) (all flights)
American Airlines "American"
Alert Me

Aircraft	Boeing 737-800 (twin-jet) (B738/Q - track or photos)
Origin	Terminal A / Gate 32 / Newark Liberty Intl (KEWR - track or info)
Destination	Terminal 4 / Gate 42B / Los Angeles Intl (KLAX - track or info)
	Other flights between these airports
Route	ZIMMZ Q42 BTRIX Q480 AIR J80 VHP J80 MCI J24 SLN J102 ALS J44 RSK J64 PGS RIIVR2 (Decode)
Date	2011年 12月 08日 (Thursday)
Duration	5 hours 43 minutes
Progress	20 minutes left 5 hours 23 minutes
Status	En Route (2,284 sm do 8 sm to go)
Distance	Direct: 2,451 sm Pl d: 2,458
Fare	$51.99 to $3,561 , average: $241.96 (airline insight)
Cabin	First: Dinner / Economy: Food for sale
	Scheduled 7-day Average Actual/*Estimated*
Departure	06:15PM EST 07:08PM EST 06:53PM EST
Arrival	08:33PM PST 09:17PM PST *09:36PM PST*

6:15 PM

8:33 PM

ORBITZ

American Airlines # 119

Leg 1: In Transit

Departs: Newark (EWR) View real-time airport conditions at]

Gate: 32

Scheduled Estim

6:22p Dec 8 - **6:32p** Dec 8

Arrives: Los Angeles (LAX) View real-time airport conditions

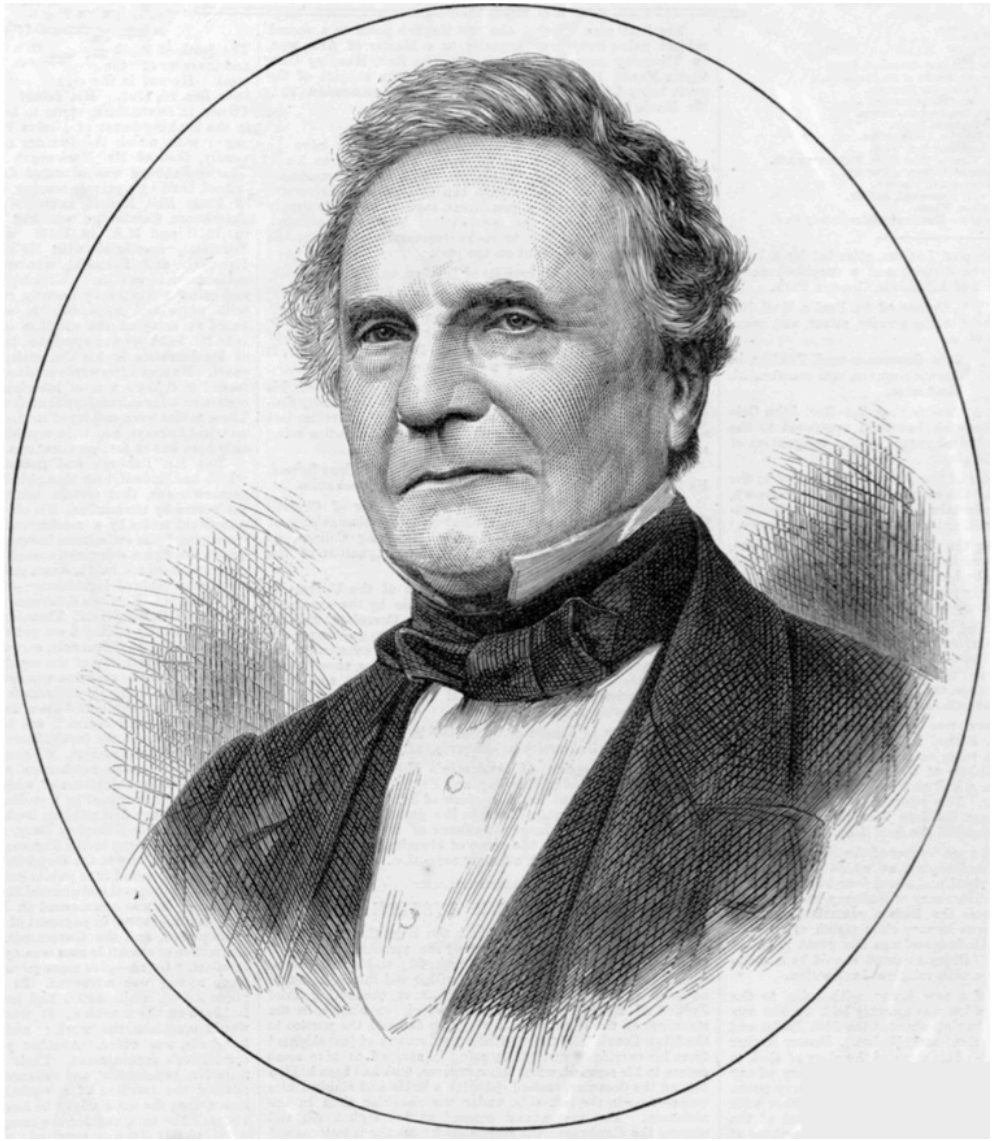
Gate: 42B

Scheduled	Estimated	Actual
9:54p Dec 8	**9:47p** Dec 8	

6:22 PM

9:54 PM

slide credit: Luna Dong and Divesh Srivastava



"Pray, Mr. Babbage, if you put into the machine wrong figures, will the right answers come out?"

Charles Babbage, from *Passages from the Life of a Philosopher*

Dealing with software errors

- program analysis
- language features
- code reviews
- formal verification

… and

testing

Key idea:
Testing for data-intensive systems

- Identifying system failures caused by well-formed but incorrect data
- Using application-specific execution information
- Integrating into the system usage workflow

Bringing testing to the data domain

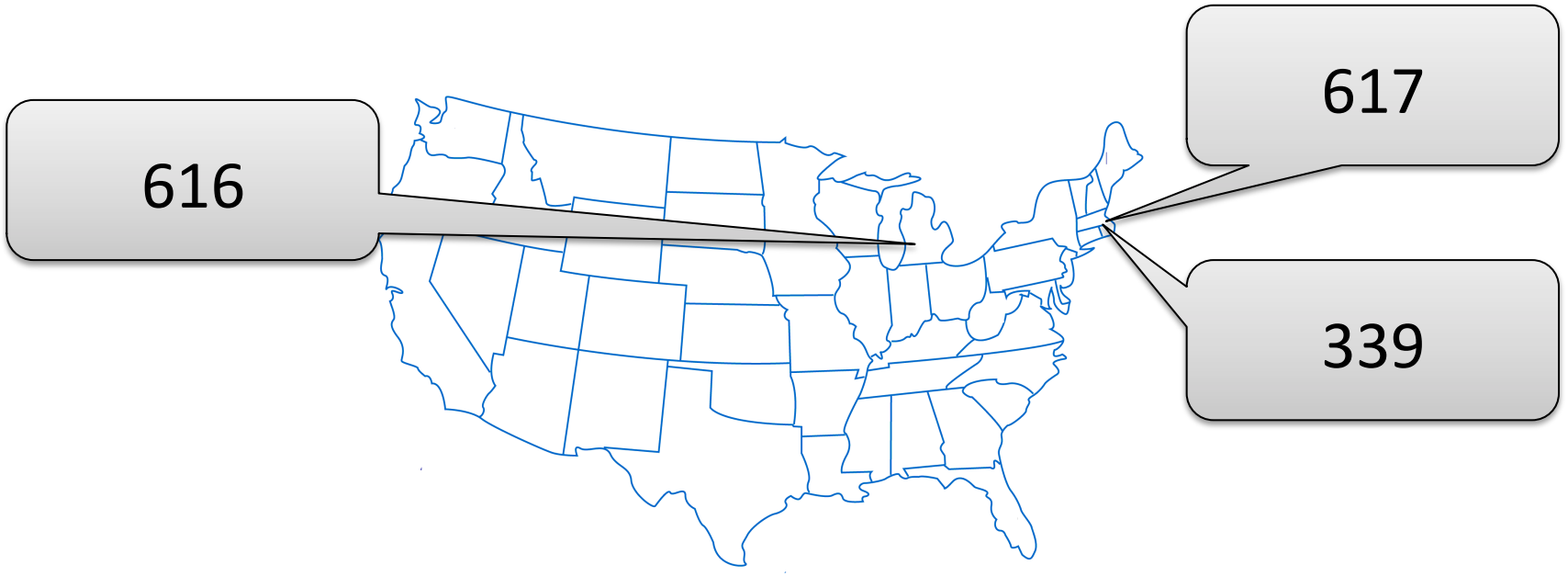
How is data testing different?

System administrators and users are not software engineers

- Data semantics
- Test query generation
- Timeliness
- Unobtrusive and precise interface

Data semantics

- Valid data can span multiple orders of magnitude
  - thwarts statistical outlier detection
- Semantics of nearly identical data differ vastly:



Test query generation

- Administrators and users have not (yet) bought into testing
- Manually written tests will come
  - developers can ship these with application
- But automatic generation is needed now
  - mine queries from code
  - record historical queries
  - adapt related work on database-use test generation

Timeliness

- Data testing is a runtime activity
- Administrators and users don’t troubleshoot unless something goes wrong
- Learning about an error sooner means error has less impact

Unobtrusive and precise user interface

- Administrators and users may not understand test outcomes
- Tests must integrate into workflow
- Results must link to
  - actions that caused failure, or
  - data values relevant to failure

Example scenario: car dealership

carID	make	model	year	inventory	cost	price
121	Nissan	Versa	2014	23	$10,990	$13,199
96	Smart	fortwo Pure	2014	21	$12,490	$14,999
227	Ford	Fiesta	2014	9	$13,200	$15,799
160	Suzuki	SX4	2014	27	$13,699	$16,499
82	Chevrolet	Sonic	2014	15	$13,735	$16,499
311	KIA	Soul	2013	3	$13,300	$14,699
319	KIA	Soul	2014	22	$13,900	$16,999
286	Toyota	Yaris	2013	1	$13,980	$15,199
295	Toyota	Yaris	2014	11	$14,115	$16,999
511	Mercedes	C-Class	2014	21	$35,800	$45,999
513	Mercedes	R-Class	2014	7	$52,690	$62,899
799	Maserati	Quattroporte	2014	8	$102,500	$122,999
808	Maserati	GranTurismo	2014	12	$126,500	$149,999

- Manager wants to put cars between $10K and $15K on a 30%-off sale

```
UPDATE   cars SET price=0.7*price
WHERE    price BETWEEN 10000 AND 150000
```

Car dealership challenges

- Data semantics

  reducing all prices by 30% seems like a valid change

- Test query generation

  manager doesn't know about writing tests

- Timeliness

  manager doesn't know to run tests

  reporting delay causes financial losses

- Unobtrusive and precise interface

  problem cannot be reported as "test failed"

Continuous Data Testing (CDT)

- Data semantics and test query generation
  - mines source code for queries
  - allows manually written and history-mined tests
- Timeliness
  - run tests continuously
  - optimizations to trigger proper tests at proper times
- Unobtrusive and precise interface
  - delegates problem to system designer
  - highlights data involved in tests

Making CDT possible (optimizations)

- NaïveCDT: run tests continuously
- SimpleCDT: only run tests after updates
- SmartCDT: only relevant tests after relevant changes
- $\text{SmartCDT}_{\text{TC}}$: test compression
- $\text{SmartCDT}_{\text{IT}}$: incremental test query execution
- $\text{SmartCDT}_{\text{TC+IT}}$: all of the above

CDT effectiveness

How effective is CDT
at preventing data entry errors?

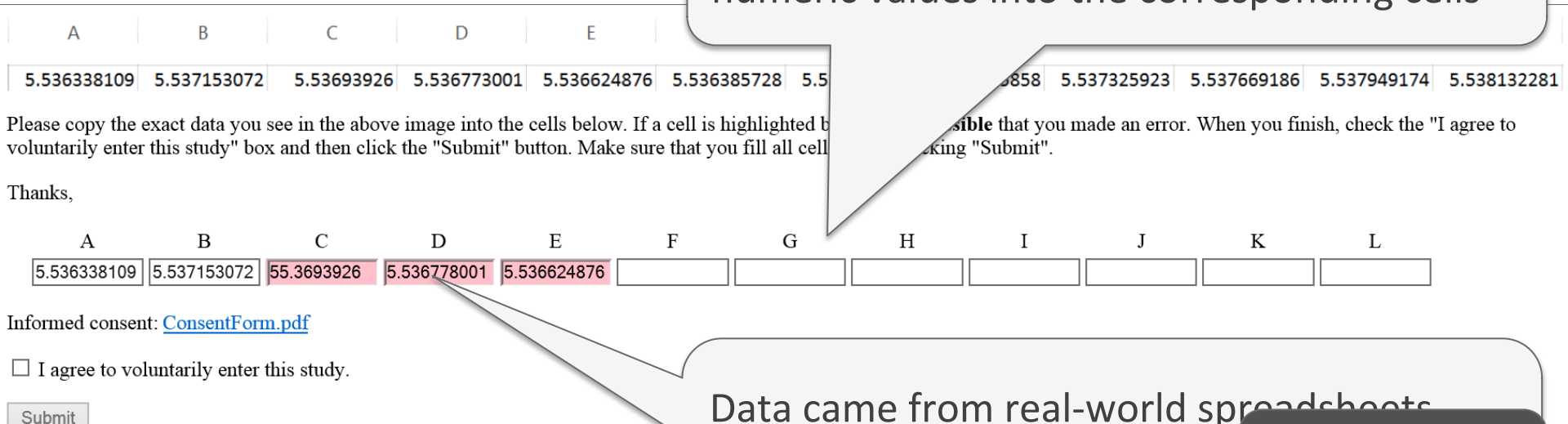
Do false positives reduce
CDT's effectiveness?

CDT's and integrity constraints' effect
on data entry speed

Crowd study

Amazon Mechanical Turk

96 distinct users were asked to copy numeric values into the corresponding cells



A	B	C	D	E
5.536338109	5.537153072	5.53693926	5.536773001	5.536624876

5.5	858	5.537325923	5.537669186	5.537949174	5.538132281

Please copy the exact data you see in the above image into the cells below. If a cell is highlighted b **sible** that you made an error. When you finish, check the "I agree to voluntarily enter this study" box and then click the "Submit" button. Make sure that you fill all cell king "Submit".

Thanks,

A	B	C	D	E	F	G	H	I	J	K	L
5.536338109	5.537153072	55.3693926	5.536778001	5.536624876							

Informed consent: ConsentForm.pdf

☐ I agree to voluntarily enter this study.

Submit

Data came from real-world spreadsheets from data.gov and the FUSES

Group 1: control

No highlighting. Submitting errors allowed.

Group 2: CDT

Data involved in failing tests highlighted. Submitting errors OK.

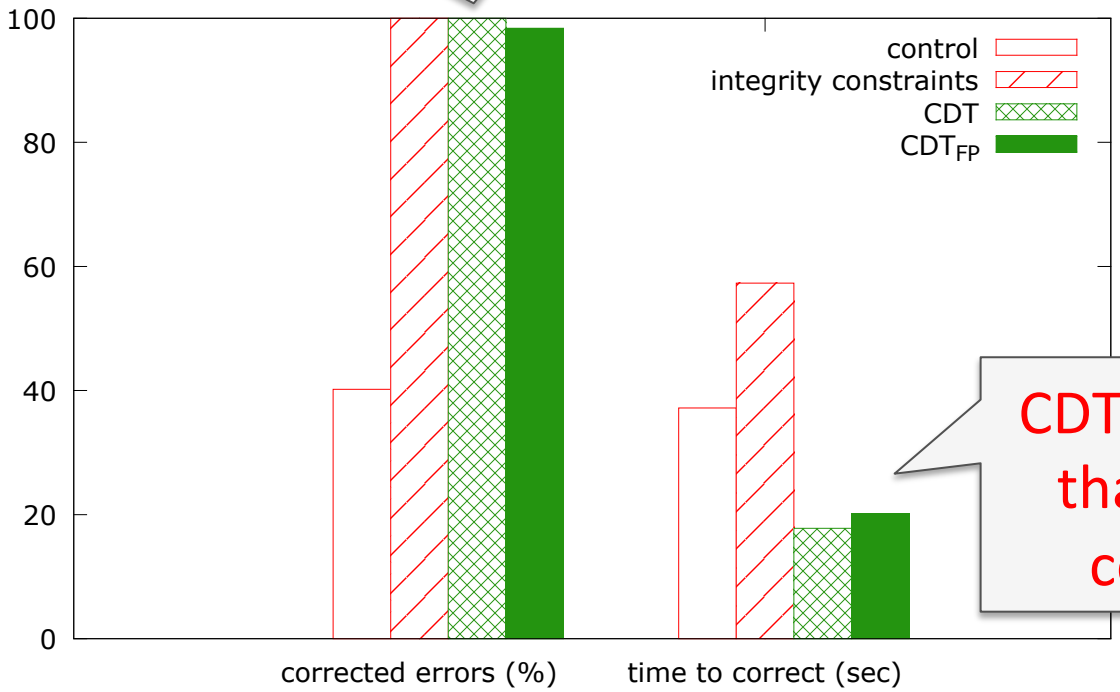
Group 3: CDT with false positives

Data involved in failing test and 40% extra data highlighted. Submitting errors OK.

Group 4: integrity constraints

Data involved in failing integrity constraints highlighted. No submitting errors.

CDT, even with false positives, successfully prevented data errors

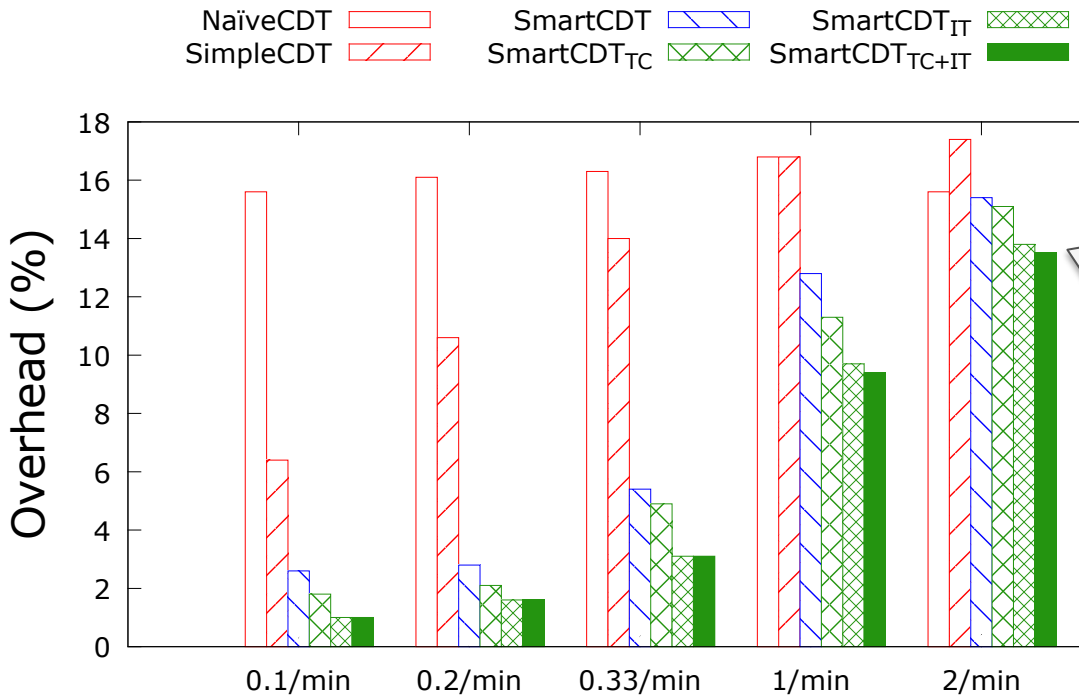


CDT much faster than integrity constraints

group	total entries	errors: total		errors: corrected		errors: submitted		time: per task	time: to correct error
control	1,209	82	6.8%	33	40.2%	49	59.8%	112 sec.	37.2 sec.
CDT	1,097	67	6.1%	67	100 %	0	0 %	126 sec.	17.8 sec.
CDT$_{FP}$	909	63	6.9%	62	98.4%	1	1.6%	97 sec.	20.1 sec.
integrity constraints	1,083	50	4.6%	50	100 %	0*	0* %	154 sec.	57.3 sec.

CDT efficiency

CDT's effect on performance of performance-intensive applications



Even with frequent updates, overhead is manageable

more frequent updates = more overhead

Related work

- Generating tests for systems that use databases

Chays, Shahid, and Frankl. Query-based test generation for database applications. DBTest 2008.

Khalek and Khurshid. Systematic testing of database engines using a relational constraint solver. ICST 2011.

Li and Csallner. Dynamic symbolic database application testing. DBTest 2010.

Pan, Wu, and Xie. Database state generation via dynamic symbolic execution for coverage criteria. DBTest 2011.

Pan, Wu, and Xie. Generating program inputs for database application testing. ASE 2011.

Related work

- Generating tests for systems that use databases
- Continuous testing

Saff and Ernst. Reducing wasted development time via continuous testing. ISSRE 2003.

Saff and Ernst. An experimental evaluation of continuous testing during development. ISSTA 2004.

Related work

- Generating tests for systems that use databases
- Continuous testing
- Effects of continuous feedback

Katzan Jr. Batch, conversational, and incremental compilers. The American Federation of Information Processing Societies 1969.

Muslu, Brun, Holmes, Ernst, and Notkin. Speculative analysis of integrated development environment recommendations. OOPSLA 2012.

Boekhoudt. The big bang theory of IDEs. Queue 2003.

Related work

- Generating tests for systems that use databases
- Continuous testing
- Effects of continuous feedback
- Errors in spreadsheets

Badame and Dig. Refactoring meets spreadsheet formulas. ICSM 2012.

Barowy, Gochev, and Berger. CheckCell: Data debugging for spreadsheets. OOPSLA 2014.

Hermans and Dig. BumbleBee: A refactoring environment for spreadsheet formulas. FSE Tool Demo 2014.

Related work

- Generating tests for systems that use databases
- Continuous testing
- Effects of continuous
- Errors in spreadsheet
- Data cleaning

Culotta and McCallum. Joint deduplication of multiple record types in relational data. CIKM 2005.

Domingos. Multi-relational record linkage. In Workshop on Multi-Relational Data Mining 2004.

Dong, Halevy, and Madhavan. Reference reconciliation in complex information spaces. SIGMOD 2005.

Li, Tziviskou, Wang, Dong, Liu, Maurino, and Srivastava. Chronos: Facilitating history discovery by linking temporal records. PVLDB 2012.

Kashyap and Sheth. Semantic and schematic similarities between database objects: A context-based approach. The VLDB Journal 1996.

Hernandez and Stolfo. The merge/purge problem for large databases. SIGMOD 1995.

Related work

- Generating tests for sy
  databases
- Continuous testing
- Effects of continuous
- Errors in spreadsh
- Data cleani
- Understanding errors in databases

Meliou, Gatterbauer, Moore, and Suciu. The complexity of causality and responsibility for query answers and non-answers. PVLDB 2010.

Meliou, Roy, and Suciu. Causality and explanations in databases. PVLDB 2014.

Wang, Dong, and Meliou. Data X-Ray: A diagnostic tool for data errors. SIGMOD 2015.

Khoussainova, Balazinska, and Suciu. Towards correcting input data errors probabilistically using integrity constraints. ACM International Workshop on Data Engineering for Wireless and Mobile Access 2006.

Related work

- Generating tests for systems that use databases
- Continuous testing
- Effects of continuous feedback
- Errors in spreadsheets
- Data cleaning
- Understanding errors in databases
- A truckload of automated test generation work

Contributions

- Four challenges of data testing:
  1. data semantics
  2. test generation
  3. timeliness
  4. interface
- Continuous Data Testing prototype for PostgreSQL
- Optimizations for which tests to run when
- CDT efficient and effective at preventing errors, even when low-quality tests result in false positives

https://bitbucket.org/ameli/contest