

Četiri stuba odbrane od spredšitova sa greškama

UDK: 004.94 ; 005.311.6

Prof dr Konstantin Kostić

Fakultet organizacionih nauka u Beogradu

SPIN 09, 05-06. novembar 2009. godine, Beograd, Srbija

Pregled radova sa prošlih EuSpRIG konferencija pokazuju da se oni mogu razvrstati u četiri prepoznatljive oblasti: alat, obrazovanje, ispitivanje i menadžment.

1. Uvod

Zahvaljujući svojoj lakoći korišćenja, fleksibilnosti i mogućnostima, spredšitovi se koriste kao podrška mnogim poslovnim funkcijama i često poprimaju uloge za koje bi primena nekih drugih rešenja bila suviše spora ili skupa. Kao rezultat toga, spredšitovi su tiho postali ključne komponente u procesima analiza i izveštavanja u mnogim organizacijama, uključujući kritične oblasti finansijskog izveštavanja.

Postoji velika evidencija o tome da su greške u spredšitovima uobičajene i nisu trivijalne. Količina greški u spredšitovima je srazmerna količini greški u drugim intelektualnim aktivnostima ljudi i one su uzrokovane fundamentalnim ograničenjima ljudskog intelekta, a ne samo ljudskom traljavošću. Sama opaska „budi pažljiv“ ne može eliminisati greške ili ih bar redukovati do prihvatljivog nivoa.

Dosadašnja istraživanja, koja su prezentirana u radovima na godišnjim konferencijama Evropske interesne grupe za spredšit rizik (EurSpRIG – European Spreadsheet Risk Interest Group) mogu se svrstati u četiri grupe:

Tools (alat) – metodologije i softver za izradu spredšitova

Education (obrazovanje) – podizanje svesti, obučavanje i istraživanje

Audit (ispitivanje) – metode i softverski alati za preispitivanje spredšitova

Management (menadžment) – standardi i kontrole za upravljanje razvojem

Prva slova naziva ovih oblasti čine reč TEAM (na engleskom znači tim). To je ispalo zgodno jer se time ukazuje da je timski rad ključ rešavanja spredšit problema u korporacijama: korišćenje razvojnih alata, obučavanje korisnika radi podizanja svesti o greškama u spredšitovima, pravilno organizovati i primenjivati procese ispitivanja spredšitova, kao i prihvatanje jasne menadžerske politike kojima će se podržati standardi i kontrole.

2. Alat

Spredšitovima se uglavnom prilazi kao intuitivnom alatu koji je varljivo lak za upotrebu i mada se naširoko koristi u organizacijama često ispadne da su dobijene informacije pogrešne. Spredšit sam po sebi ima odredjene ograničenja koja sprečavaju razvijanje kompleksnih poslovnih modela kao što i sam korisnik spredšita (kreator) može da bude nevešt u razvijanju takvog modela.

Bez svake sumnje, postoji potreba da se definiše i primeni struktuirani pristup izrade modela uz pomoć softverskih alata kojima bi se rukovodio kreator spredšita kako bi koristio savete da izbegne (ili predupredi) moguće situacije pogreški, kao i da dokumentuje svoje akcije.

Dosadašnji radovi na skupovima uključivali su i one koji zagovaraju korišćenje programskih jezika kojima bi se „kodirala“ logika modela i tek onda konstruisao interfejs spredšita u redovima i kolonama zajedno sa svojom uključujućom logikom. Jedan autor zagovara korišćenje matematičke grane poznate kao „teorija kategorija“ [13]. Teorija kategorija, kao logika, je alat za proučavanje matematičkih i računarskih koncepcata koja je više koncentrisana na formu nego na sadržinu.

Mnogi učesnici skupova su isticali potrebu stvaranja metoda koje se zasnivaju na principima softverskog inženjerstva. Oni tvrde da primena principa softverskog inženjerstva – što bi činilo spredšit inženjerstvo – ima potencijal da poveća produktivnost spredšit programera, smanji učestanost i težinu greški u spredšitovima, unapredi manipulativnost spredšitovima u vremenu.

Rajalingham, Chadwick and Knight [14] nude alat koji kreira Džeksonove strukture radi dobijanja grafičkog tipa opisa spredšita tokom razvoja¹. Ovo omogućava laku proveru logike i dopušta reverzni inženjer-

¹Spredšit modeli se mogu prikazati u formi identičnoj dijagramu strukture podataka - Džeksonova struktura koja se koristi u softverskom inženjerstvu.

ring spredšta u njegovoj „Jackson“ strukturi. Dakle, svaki spredšti model može se definisati pomoću njegovog kanoničnog oblika koji će ostati netaknut čak i kad se forma spredšta promeni kozmetički.

Na tržištu se nalazi veliki broj spredšta softvera. Lstu najpoznatijih spredšta softvera pogledajte na sajtu http://en.wikipedia.org/wiki/List_of_spreadsheet_software. Na skupovima su ponudjene i nove spredšti alternative. Kombinacija najnovijih objektnih tehnologija sa novim pristupima nude istinske alternative spredštovima koje idu u pravcu zadovoljavanja do sada identifikovanih otpora prema spredštovima. Na primer XBRL (Extensible Business Reporting Language) je jezik za elektronsku komunikaciju finansijskih podataka koji je revolucionizovao finansijsko izveštavanje širom sveta. Koristi se za pripremu, analizu i komunikaciju finansijskih informacija. XBRL je aplikacija XML-a za poslovne informacije i koristi strukturu da opiše podatke, čini ih odmah upotrebljivim, interaktivnim i inteligentnim.

Što se tiče dokumentovanja spredštova, dobra dokumentacija:

- Opisuje suštinu i način korišćenja modela tako da dopušta i drugima da koriste model, ali koristi i kada se ponovo aktivira model posle nekog vremena nekorišćenja
- Smanjuje rizik da model prestane da se koristi zato što samo jedan ili par ljudi zna kako da se koristi i imaju poverenja u njega
- Smanjuje broj iritirajućih prekida osobi koja je preuzeila dužnost da dalje razvija model.

U dokumentaciji treba da se navede: čemu služi model, šta on radi, kako to radi, koje su prepostavke uključene u njegov dizajn, koje se konstante koriste i gde se one nalaze, ko i kada je razvio model, ko i kada je izvršio modifikacije modela posle njegovog stavljanja u upotrebu, postojanje makroa i njihova svrha.

Korišćenje Excel-ovih komentara na sistematičan način može da predstavlja jednu efikasnu metodu dokumentovanja spredštova.

3. Obrazovanje

Da bi se projektovalo i primenilo jedno efikasno spredšti rešenje, važno je da se poseduje poznavanje posla kako bi se podaci pretvorili u korisne informacije. Poznavanje tehnike se stiče treningom. Ako tehničko znanje ne uzima u obzir poznavanje posla, onda rešenja neće biti optimalna.

Najefikasniji menadžment spredšta sistema u jednoj organizaciji postiže se treningom koji se bazira na rešavanju konkretnih problema (task-based management). Taj pristup obučava korisnika kako da rešava konkretni problem spredšta alatima za razliku od pristupa izučavanja spredšta alata i tehnika nezavisno od bilo kog konkretnog problema.

Treneri koji koriste «task-based» pristup imaju iskustva u načinu kako se spredštovi stvarno koriste u realnom svetu. Oni kombinuju svoje iskustvo sa klijentovim poznavanjem poslovnog okruženja. To je bolje od suvoparnog teoretisanja o tome kako bi trebalo koristiti spredštove.

Na kraju treninga polaznik je:

- ✓ Kompletirao svoj zadatak
- ✓ Postavio sistem za buduće zadatke
- ✓ Video je rešenja za neke od svojih problema
- ✓ Radio je na aktuelnom spredštu

Polaznik je učio, kako mehaničke tehnike vezane za zadatak, tako i rezon korišćenja tih pojedinačnih tehnika u datim situacijama.

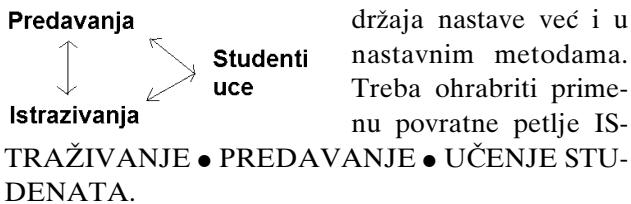
«Task-based» trening nije jedini tip treninga i daleko od toga da je najpopularniji. Mnogo rasprostranjeniji tip treninga je uopštena obuka upoznavanja sa spredšt programskim paketom pošto se medju polaznicima obuke obično nalaze ljudi koji spredšti koriste u različite svrhe.

Treći tip «obuke» je najčešći, a obuhvata korisnike koji nikada nisu imali bilo kakvu formalnu obuku o spredštovima. Oni su naučili korišćenje spredštova na poslu od svojih prethodnika ili kolega, ili su potpuno samouki.

Kvalitetnu obuku korišćenja spredštova daje Fakultet organizacionih nauka u Beogradu i još par drugih fakulteta. Obuka za rad sa spredštovima na fakultetima pati od lošeg pristupa spredštovima:

- Spredštovi se predstavljaju kao prosti iskazi jednostavnih računovodstvenih modela
- Spredštovi su laki za učenje i njihov intelektualni sadržaj je ograničen
- Student je učio o spredštovima još pre univerziteta i time stekao pogrešne navike koje se ne ispravljaju na fakultetu.

Možda je najveći problem da se ubede nastavnici da obuka za rad u spredštovima zahteva kompletno drugačiji pristup koji se ne zasniva samo na promeni sa-



4. Ispitivanje

Postoje mnogobrojni tipovi testiranja od kojih većina nije dovoljna da smanji greške do tačke koja obezbeđuje da nema značajnih greški u spredštu. Nabroju- ĉu neke od tipova.

- 1. Testiranje tokom razvoja spredšta** – Testiranje zahteva, testiranje jedinica (modula), integralno testiranje.
- 2. Posmatranje spredšta** – Pregled tabela u spredštu da se stekne utisak logičnosti.
- 3. Softver za skeniranje greški** – softver koji pretražuje spredšit tražeći greške na osnovu pravila za pro- nalaženje koja su ugradjena u softver. Alati za au- tomatsko skeniranje greški slični su spel-čekeru i gramar-čekeru u programima za obradu teksta. Excel ima ugradjeni alat za pronalaženje greški pod komandoma Tools, Error Checking. Ovaj alat je ve- oma jednostavan ali ograničen. Postoji nekoliko komercijalnih alata koji su raspoloživi za ispitivanje spredštova: SpACE (Spreadsheet Audit for Cu- stoms & Excise), OAK (Operis Analysis Kit), Spre- adsheet Detective and Spreadsheet Professional i drugi. Mnogo sofisticiraniji skeneri se mogu naći na tržištu. Web strana sa listom takvih proizvoda je <http://www.sysmod.com/sslinks.htm#auditing>.
- 4. Auditing** – opcija koja se kod Excela aktivira komandama Tools, Formula Auditing. Ova komanda omogućava da izaberemo vezu formule sa prethod- nim celijama od kojih zavisi rezultat formule, odno- sno sa svim celijama kojima vrednost posmatrane celije služi kao ulazni podatak. Jedan od problema sa istinskim oditingom je pitanje koji deo spredšta da se ispituje. Najočigledniji kandidat za ispitivanje je „najriskantniji deo“ koji uključuje kompleksne formule i linkove medju radnim listovima radne knjige. Medjutim, dok su greške verovatnije kod ovog tipa formula nego što je to slučaj kod običnih formula, po zakonu verovatnoće je verovatno više greški kod običnih formula nego kod kompleksnih formula. (običnih formula ima više). Na tržištu po- stoje automatizovani auditing softveri za spredšto- ve, koji su veoma kompleksni ali i skupi.
- 5. Testiranje izvršenjem** – tester provera nekoliko se- tova ulaznih vrednosti da bi video da li spredšit proizvodi (ili ne proizvodi) korektne rezultate. Pri

tome se mogu koristiti tipične i ekstremna vredno- sti (paranoidno testiranje). Ovde se postavlja prob- lem kako znati da su dobijene vrednosti tačne. Najbolje je kada postoji mogućnost da se rezultu- juće vrednosti uporeduju sa unapred poznatim vrednostima. Dobro bi bilo da postoji matematički model koji može da da vrednosti koje treba očekivati kao rezultat i u spredštu. Bez takvih vrednosti za uporedjivanje testiranje izvršenjem gubi smisao.

- 6. Inspekcija logike** – inspektor ispituje sve celije sa formulama, tražeći greške. Ovde se preporučuje timska inspekcija iz praktičnih razloga: pojedinac obično pronadje manje od polovine svih greški. Pri tome se ne može zahtevati redukcija svih greški jer je to nemoguće. Zato se postavlja jedna razumna granica u skladu sa značajem spredšta za donoše- nje poslovnih odluka.

Proces inspekcije treba da ima sedam koraka:

1. Planiranje. Priprema materijala, dobijanje sauče- snika i izrada rasporeda sastanaka.
2. Razmatranje sadržaja sastanaka. Predstavljanje softvera, raspodela uloga, opis procesa.
3. Priprema. Ispektori, radeći pojedinačno, ispituju spredšit. Cilj nije da se otkriju greške već da se raz- umiju softverski moduli. Medjutim, mnoge meto- de inspekcije otkriju greške već u ovoj fazi.
4. Sastanak inspektora. Zadatak je da se otkriju i ob- jasne greške. Ne diskutuje se ništa izvan toga. Sa- stanak mora da bude razumno kratak da ne bi opa- la obazrivost.
5. Unapredjenje procesa. Inspekcija mora da proiz- vede povratnu informaciju za vodič procesa in- spekcija firme. Svaka inspekcija mora da generiše statistiku o utrošenom vremenu, nadjenim greška- ma i ozbiljnosti greški.
6. Prerada. Popravka softvera se vrši posle sastanka.
7. Praćenje. Osigurati da su promene učinjene pravil- no.

5. Menadžment

Postoji mnogo, neki kažu suviše mnogo, pitanja koja se odnose na menadžment spredštova na koja treba obratiti pažnju politikom organizacije. Postoje pitanja izbora i prihvatanja seta kontrola koja treba razmotri- ti u regularnim intervalima vremena. To mogu da bu- du kontrole verzija spredštova, uvida i samo-ispitiva- nja tokom izrade spredštova, timsko preispitivanje i „pročešljavanje“ spredšit modela, politika dokumento- vanja, korišćenje „proverenih“ modela kao templejta, kao i beleženje uobičajenih greški i njihovo čuvanje na za to utvrđenim mestima kao bazama korporativ- nog znanja.

Da bi se ostvario samo deo tih zahteva potrebno je obezbediti podršku menadžmenta preduzeća višeg nivoa i stvoriti strategiju za uspostavljanje dobre prakse zasnovane na razuman izbor standarda.

Takov standard se može naći u pristupu CobiT (Control Objectives for Information and related Technology). CobiT je set alata koji pomaže poslovnim menadžerima da razumeju i upravljaju rizikom vezanim za uvodjenje nove tehnologije. Takođe, on demonstrira zainteresovanima koliko dobro je urađen taj posao. CobiT je zasnovan na međunarodnoj najboljoj praksi IT menadžmenta.

CobiT obezbeđuje generički okvir za sve glavne IT procese. On ne spominje posebno spredštove. Međutim, taj generički okvir se može primeniti na sva IT razvojnih rešenja od celokupnog ERP sistema do relativno jednostavnog spredšta.

Na primer, za kontrolu procesa «Pribavljanje i održavanje softvera» navodi se:

«Kontrola IT procesa pribavljanja i održavanja aplikacionog softvera, koji zadovoljava poslovne potrebe automatizacije funkcija radi efikasne podrške poslovnim procesima, obezbedjena je definicijama funkcionalnih i operativnih zahteva i faznom primenom sa jasnim izvršenjima i uzima u obzir:

- Funkcionalno testiranje i prihvatanje
- kontrole aplikacije i zahteve obezbedjenja
- zahteve dokumentovanja
- životni ciklus softverske aplikacije
- informacionu strukturu preduzeća
- metodologiju životnog ciklusa sistema razvoja
- interfejs korisnik – mašina
- prilagodjavanje paketa

Ovo je podržano sa 17 detaljnih ciljeva kontrole pokrivajući:

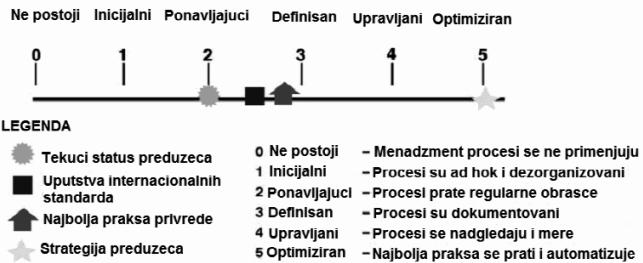
- Metode projektovanja
- Glavnih promena postojećeg sistema
- Odobrenje projekta
- Definisanje i dokumentovanje potrebnih fajlova
- Specifikacija programa
- Projekat kolekcije izvornih podataka
- Definisanje ulaznih zahteva i dokumentacije
- Definisanje interfejsova
- Interfejs korisnik – mašina
- Definisanje potrebnih procesa i dokumentacije
- Definisanje potrebnih izlaza i dokumentacije
- Mogućnost kontrole

- Pristupačnost kao ključni faktor dizajna
- Mere IT integriteta u aplikaciju softverskog programa
- Testiranje softverske aplikacije
- Korisnička uputstva i drugi materijali za podršku
- Ponovna procena sistemskog projekta

Pristup CobiT se zasniva na „Maturity“ modelu za mogućnosti razvoja softvera koji je definisan od strane firme „Software Engineering Institute“. «Maturity» model nudi metodu za kvantitativno iskazivanje stepena IT kontrolnih procedura na skali od 0 (ne postoji) do 5 (otmiziran). Menadžment koristi «Maturity» model da mapira tekuci status:

- svoje organizacije
- najbolje prakse ili opšte stanje prakse u svojoj grani
- internacionalnih standarda

i da odredi gde organizacija želi da bude u odnosu na te nivoe.



Slika 1: CoBIT Maturity Model

Kontrola procesa razvoja i održavanja spredšit modela i aplikacija koje zadovoljavaju poslovne potrebe da se dobiju tačni poslovni modeli i analize bez greški kako bi se efikasno podržali poslovni procesi, obezbeđuje se definisanjem specifičnih stavova funkcionalnih i operativnih zahteva kao i faznom implementacijom sa jasnim izvršenjima i uzima u obzir

- metode projektovanja
- zaštitu i zahteve čuvanja podataka
- testiranje i prihvatanje
- zahteve dokumentovanja

6. Zaključak

Da bi se obezedio svrsishodan spredšit informacioni sistem u jednoj organizaciji neophodno je unapredjivati četiri oblasti intelektualnog delovanja:

ALAT – usvojiti metodologiju strukturnog opisa podataka i njihovih veza kao osnovu opisa poslovnog modela koji se potom razvija u spred-

šitu; odabratи adekvatan softver za spredšit; obezbeditи dokumentovanje spredšita;
OBRAZOVANJE – Odabratи najpovoljniji način obuke korisnika (kreatora) spredštova i uspostaviti svrshodan sistem permanentnog obravozanja u sprezi sa istraživanjima.
ISPITIVANJE - uspostaviti sistem ispitivanja (oditinga) spredštova uz primenu najsavremenijih oditing alata (softvera);
MENADŽMENT – usvojiti standarde i kontrole u vezi sa upravljanjem verzijama, čuvanjem, korišćenjem templejta, merenjem performansi spredšit sistema i kontinualnim unapredjenjem.

LITERATURA

- [1] Panko R Spreadsheet Errors: What We Know. What We Think We Can Do. Proceedings of 1st International Symposium on Spreadsheet Risks, EUSPRIG2000, Greenwich, UK, July 2000. <http://arxiv.org/abs/0802.3457>
- [2] Rajalingham K, Chadwick D, Knight B: Classification of Spreadsheet Errors Proceedings of 1st International Symposium on Spreadsheet Risks, EUSPRIG2000, Greenwich, UK, July 2000. <http://arxiv.org/abs/0805.4224>
- [3] Hawker A: Building Financial Accuracy into Spreadsheets, Proceedings of 1st International Symposium on Spreadsheet Risks, EUSPRIG2000, Greenwich, UK, July 2000. <http://arxiv.org/abs/0805.4219>
- [4] Ayalew Y, Clermont M, Mittermeir R: Detecting Errors in Spreadsheets, Proceedings of 1st International Symposium on Spreadsheet Risks, EUSPRIG2000, Greenwich, UK, July 2000. <http://arxiv.org/abs/0805.1740>
- [5] Rajalingham K, Chadwick D, Knight B: A Structured Methodology for Spreadsheet Modelling. Proceedings of 1st International Symposium on Spreadsheet Risks, EUSPRIG2000, Greenwich, UK, July 2000. <http://arxiv.org/abs/0805.4218>
- [6] Butler R Risk Assessment for Spreadsheet Developments (HM Customs and Excise) Proceedings of 1st International Symposium on Spreadsheet Risks, EUSPRIG2000, Greenwich, UK, July 2000. <http://arxiv.org/abs/0805.4236>
- [7] Chan H Q Chen Y: Visual Checking Of Spreadsheets Proceedings of 1st International Symposium on Spreadsheet Risks, EUSPRIG2000, Greenwich, July 2000. <http://arxiv.org/abs/0805.2189>
- [8] Cleary P M: How Important Are Spreadsheets To Organisations? Proceedings of 1st International Symposium on Spreadsheet Risks, EUSPRIG2000, Greenwich, UK, July 2000.
- [9] Chadwick D Stop the Subversive Spreadsheet. Proceedings of 1st International Symposium on Spreadsheet Risks, EUSPRIG2000, Greenwich, July 2000. <http://arxiv.org/abs/0712.2594>
- [10] Chadwick D., Sue R.: Teaching Spreadsheet Development Using Peer Audit and Self-Audit Methods for Reducing Errors, Proceedings of 2nd International Symposium on Spreadsheet Risks, EUSPRIG2001, Amsterdam, Holland, July 2001. <http://arxiv.org/abs/0801.1514>
- [11] Butler R J: Applying the Cobit® Control Framework to Spreadsheet Developments Proceedings of 2nd International Symposium on Spreadsheet Risks, EUSPRIG2001, Amsterdam, Holland, July 2001. <http://arxiv.org/abs/0801.0609>
- [12] O'Beirne P: Euro Conversion in Spreadsheets Proceedings of 2nd International Symposium on Spreadsheet Risks, EUSPRIG2001, Amsterdam, Holland, July 2001
- [13] Paine J: Safer Spreadsheets With Model Master Proceedings of 2nd International Symposium on Spreadsheet Risks, EUSPRIG2001, Amsterdam, Holland, July 2001. <http://arxiv.org/abs/0801.3690>
- [14] Rajalingham K, Chadwick D, Knight B: An Evaluation of the Quality of a Structured Spreadsheet Development Methodology Proceedings of 2nd International Symposium on Spreadsheet Risks, EUSPRIG2001, Amsterdam, Holland, July 2001. <http://arxiv.org/abs/0801.1516>
- [15] Raffensperger J: New Guidelines for Writing spreadsheets Proceedings of 2nd International Symposium on Spreadsheet Risks, EUSPRIG2001, Amsterdam, Holland, July 2001.
- [16] Ettema H, Janssen P, de Swart J: Assurance By Control Around Is A Visible Alternative To The Traditional Approach Proceedings of 2nd International Symposium on Spreadsheet Risks, EUSPRIG2001, Amsterdam, Holland, July 2001. <http://arxiv.org/abs/0801.4775>
- [17] Knight D.; A Real alternative to spreadsheets Proceedings of 2nd International Symposium on Spreadsheet Risks, EUSPRIG2001, Amsterdam, Holland, July 2001

- [18] Nixon D, O'Hara M: Spreadsheet Auditing Proceedings of 2nd International Symposium on Spreadsheet Risks, EUSPRIG2001, Amsterdam, Holland, July 2001
- [19] Chadwick D: Training Gamble Leads To Corporate Grumble Proceedings of 3rd International Symposium on Spreadsheet Risks -the Hidden Corporate Gamble, EUSPRIG 2002, Cardiff, Wales, UK, July 2002
- [20] Banks D., Monday A.: Interpretation as a Factor in Understanding Flawed Spreadsheets Proceedings of 3rd International Symposium on Spreadsheet Risks -the Hidden Corporate Gamble, EUSPRIG 2002, Cardiff, Wales, UK, July 2002.
<http://arxiv.org/abs/0801.1856>
- [21] Grossman T.: Spreadsheet Engineering: A Research Framework Proceedings of 3rd International Symposium on Spreadsheet Risks - the Hidden Corporate Gamble, EUSPRIG 2002, Cardiff, Wales, UK, July 2002.
<http://arxiv.org/abs/0711.0538>
- [22] Clermont M., Hanin C., Mittermeir R.: A Spreadsheet Auditing Tool Evaluated In An Industrial Context Proceedings of 3d International Symposium on Spreadsheet Risks - the Hidden Corporate Gamble, EUSPRIG 2002, Cardiff, Wales, UK, July 2002.
<http://arxiv.org/abs/0805.1741>
- [23] Chadwick D., Knight B.: The Subversive Spreadsheet Proceedings of 3d International Symposium on Spreadsheet Risks – the Hidden Corporate Gamble, EUSPRIG 2002, Cardiff, Wales, UK, July 2002.
- [24] Butler R.: Losing At Spreadsheet Roulette Proceedings of 3rd International Symposium on Spreadsheet Risks - the Hidden Corporate Gamble, EUSPRIG 2002, Cardiff, Wales, UK, July 2002
- [25] Croll G.: A Typical Spreadsheet Audit Approach Proceedings of 3d International Symposium on Spreadsheet Risks - the Hidden Corporate Gamble, EUSPRIG 2002, Cardiff, Wales, UK, July 2002. <http://arxiv.org/abs/0712.2591>