



דם, יזע ודו"ח ביצועים

הרצה על בדיקה פונקציונאלית מסתיימת בתוצאה סופית ברורה: עבר או נכשל. דיווח על בדיקות ביצועים (performance) לעומת זאת, זה משהו אחר לגמרי.

כל זה הוא רק הקדמה לנושא האמיתי של הטור.

איך מדווחים על בדיקות ביצועים?

עקרונית, מה הבעיה? מציגים את התוצאה ומציינים אם היא עברה או נכשלה. אבל מעבר לתוצאה אנו גם רוצים להראות כמה אנחנו קרובים למטרה, ולצורך כך צריך דיווח שמצד אחד אינו מסורבל ומצד שני מעביר למתבונן תמונה מלאה יותר של המצב. אפשר למשל להשתמש בטבלה:

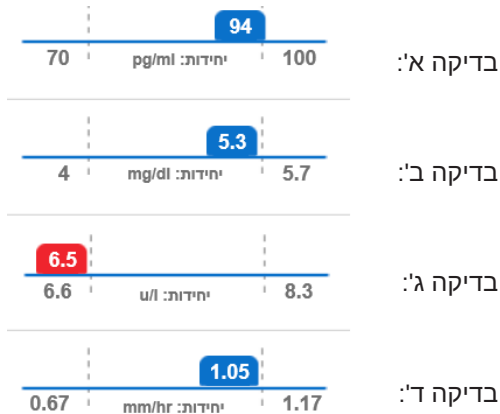
תוצאה	מינימום (limit)	מטרה (target)	דרישה
101	100	120	קצב עיבוד תמונות (fps)

זה אכן מספק את המידע, אבל כיוון שיש בכל מוצר יותר ממדידה אחת, הרי שנגיע לטבלה גדולה מלאה מספרים, וקשה יהיה להבחין בהסתכלות מהירה איפה מתחבאת בעיה. אפשר להשתמש בצבעים על מנת לשפר את הקריאות:

תוצאה	מינימום (limit)	מטרה (target)	דרישה
101	100	120	קצב עיבוד תמונות (fps)
8.55%	10%	7%	CPU utilization
1.34W	1.9W	1.5W	צריכת הספק

נראה לכם? למה קצב עיבוד תמונות קיבל אותו צבע כמו המדידה של ה-CPU? הרי אחד ממש קרוב לקצה והשני בכלל לא... מצד שני, האם יהיה מוצדק לצבוע את קצב עיבוד התמונות באדום? אם נעשה ככה, אז באיזה צבע נסמן כישלון בבדיקה? אדום חזק יותר? ומה ירוק אומר? עד מתי תוצאה היא ירוקה ומתי תהפוך לכתומה? בקיצור, צבעים קצת עוזרים, אבל גם מכניסים בעיות משלהם.

העניין הסתובב לי הרבה זמן בראש. גם כשרופא המשפחה שלח אותי לסבב בדיקות הדם השנתי, אותו אני מקפיד לעשות בערך פעם בשלוש שנים. התוצאות שחזרו מהמעבדה היו רשימה של כמה עשרות מספרים שאני מבין בהם פחות מתרגול במרק פירות. אבל... וכאן מגיע הקשר לענייננו: למרות שאינני רופא, יכולתי לומר מיד איפה יש תוצאה טובה, איפה גבולית ואיפה משהו שכדאי לשאול עליו את הרופא. ככה זה נראה:



מיד נדלקה נורה בראשי: הרי זה בדיוק האתגר בדיווח תוצאות בבדיקות ביצועים! אולי אפשר לגנוב את השיטה וליישם גם אצלנו? לקחתי כמה נתונים ושיחקתי קצת עם PowerPoint, והרי התוצאה:

קודם כל הגדרה:

תחת השם "בדיקות ביצועים" אני מכניס, לצורך המאמר הזה, כל בדיקה שהתוצאה שלה היא מדידה של משהו; ערך מספרי שיכול לנוע בתוך תחום מסוים ועדיין להיחשב כתקין. זה יכול להיות מדידת צריכת זרם, כמות המשתמשים שאתר רשת מסוגל לטפל בה בו-זמנית, מהירות שליפת נתונים מהדיסק, וכו' וכו'. למעשה: כל מדידה של דרישה לא-פונקציונאלית.

הקושי הראשון של בדיקות ביצועים הוא לקבוע מה טווח התוצאות שיחשב כ"עובר". זה מתחיל כבר בשלב הגדרת הדרישות. הרבה פעמים הדרישה היא משהו בסגנון הבא: "זמן אחזור נתון מבסיס הנתונים לא יעלה על 10 מילישניות" או "המערכת תעבד קבצי וידיאו בקצב של לפחות 100 תמונות בשנייה". הגדרות כאלה הן חסרות (incomplete) כי לא ברור מה התוצאה הרצויה. ידוע רק מה התוצאה הגרועה ביותר שנמדוד ועדיין נאמר שהמוצר עבר את הבדיקה. למעשה מתחבאות כאן שתי בעיות:



מיכאל שטאל

הוא ארכיטקט בדיקות תוכנה באינטל, ישראל. במשך 17 השנים האחרונות מיכאל בדק מוצרים בתחום Wi-Fi, טלויזיות חכמות, כרטיסים גרפיים, אפליקציות ראייה ממוחשבת מבוססות על מצלמות תלת-מימד, ולאחרונה הוא עובד בקבוצה שבודקת את התוכנה שמאחורי טכנולוגיית ניהול אקטיבי ומנוע האבטחה (CSME) של אינטל. במסגרת תפקידו, מיכאל מגדיר שיטות בדיקה ומתודולוגיות עבודה, עוסק הרבה בהדרכה ולפעמים אפילו מרשים לו לבדוק משהו (שזה הכי כיף). מיכאל מציג תכופות בכנסים בארץ ובחו"ל והציג בין היתר ב-SIGIST Israel, STAR conferences, QA&Test, ובכנסים אחרים. מיכאל מלמד בדיקות תוכנה בפקולטה למדעי המחשב באוניברסיטה העברית. ניתן לראות חלק מהמצגות והמאמרים שלו באתר

www.testprincipia.com

(1) נניח שאני מריץ בדיקה ומוצא שקצב עיבוד התמונות הוא 101 תמונות לשנייה (הדרישה הייתה "לפחות 100 תמונות בשנייה"). נראה טוב, נכון? אבל האם זה אומר שאני קרוב לקצה (כלומר, עברתי את הבדיקה בעור שיני) או שהכל תקין? אם הדרישה הייתה מוגדרת היטב, היא הייתה כוללת את המטרה (נאמר: 120 תמונות לשנייה) וגם את מינימום הביצועים הנדרשים: 100 תמונות לשנייה. עכשיו, כשאני מודד 101 תמונות לשנייה, ברור שהמוצר עומד אך בקושי בדרישה.

(2) כאשר נכשלים בבדיקה רק קצת (נגיד, 99 תמונות לשנייה), יש לחץ על מנהל המוצר "להתגמש". "אוקי, זה באמת מתחת לגבול, אבל הרי אנחנו כמעט עוברים, אז אפשר להחליט שזה בסדר". אילו הדרישה המלאה הייתה עומדת מול עיני המחליטים, הם היו מודעים לעובדה שהמוצר רחוק מהמטרה ושיש כאן בעיה אמיתית.

רק לשם שלמות, אזכיר שהגדרת דרישה לא-פונקציונאלית צריכה לכלול בפרוטרוט גם את תנאי הבדיקה ומהלכה, שכן אלה משפיעים על התוצאה הנמדדת (למשל, בבדיקת עומס על המעבד: האם אנחנו מודדים Max CPU Utilization או ממוצע לאורך זמן? כמה זמן? כל כמה זמן דוגמים את עומס המעבד? מה עוד רץ על המחשב במקביל?).



“
איך לדווח תוצאות מוקדמות של בדיקות ביצועים, ובמקביל להעביר את המסר שזה תוצאות ראשוניות
”

בדרך כלל בתחום של סטיית תקן אחת. לעיתים רחוקות יותר בתחום שבין 1 ל-2 סטיות תקן וכמעט אף פעם

לא בתחום שבין 2 ל-3 סטיות תקן. בכל מקרה, המדידות יתפלגו באופן שווה פחות או יותר מעל ומתחת למוצע. אם המדידות על גרסאות חדשות נופלות בתחום של סטיית תקן אחת, אבל באופן קבוע מעל (או מתחת) למוצע, כנראה שאכן קרה שינוי. כמה פעמים המדידות צריכות להיות מוטות לכיוון מסוים על מנת שנוכל לומר שהשינוי משמעותי? זה תלוי בהתפלגות שלנו, ונכנס לתחום הנקרא "בקרת איכות סטטיסטית" (statistical process control). דוגמה לאלגוריתם המחליט מתי הטיה היא משמעותית הם הכללים של "יוסטון דיגיטל":

- 8 וורסיות עוקבות של התוכנה שבהם הנטייה היא בתחום סטיית תקן אחת – אבל לאותו כיוון
- מדידות מ-5 וורסיות עוקבות, שמתוך 4 מדידות נותרת תוצאה בין סטיית תקן 1 ל-2 מעל (או מתחת) למוצע – באותו כיוון
- מדידות מ-3 וורסיות עוקבות, שמתוך 2 מדידות נותרת תוצאה בין 2-3 סטיות תקן מעל (או מתחת) למוצע – באותו כיוון
- מקרה יחיד שבו המדידה היא מעבר ל-3 סטיות תקן מהמוצע – לאחד הכיוונים

הצצה בוויקיפדיה מראה גרפים המתארים את הכללים האלה- https://en.wikipedia.org/wiki/Western_Electric_rules.

כל זה טוב ויפה, אתם אומרים, אבל אנחנו לא יודעים מה היכולת של המערכת שלנו ואין לנו זמן להריץ מדידות חוזרות רבות על מנת לייצר את הסטטיסטיקה הזאת. לא תמיד אנחנו מריצים את הבדיקה על אותו מחשב ממש, כך שכנסנו רעש נוסף שנובע מההבדלים הקטנים בין מחשב למחשב – גם אם המפרט שלהם זהה לחלוטין. במקרים שבהם כן יש מושג על ההתפלגות, יתכן מאוד שהיא אינה נורמלית ולכן לא ברור לפי איזה כללים יקבע מתי יש שינוי אמיתי.

מה עושים? במקרה כזה אפשר להחליט מתוך הבנת המערכת איזה שינוי נראה לכם משמעותי (נניח "שינוי של 2% הוא משמעותי"). עם החלטה כזו, אפשר להגדיר כללי אצבע פשוטים יותר ל"מגמה". הנה גישה שפיתחתי עם גל שטיינר, חבר לעבודה:

- (1) "מגמה" נקבעת על פי השוואה של התוצאה שנמדדה על הגרסה הנוכחית לעומת התוצאות של הגרסה הקודמת (גרסה N-1)
 - (2) מגמה יכולה להיות: ללא שינוי, עלייה, ירידה
 - (3) אם השינוי מ-N-1 הוא מעל ל-2%, יש מגמת עלייה
 - (4) (או ירידה – תלוי בכיוון השינוי)
- אם השינוי מ-N-1 הוא קטן מ-2% אזי נבדוק מה הייתה המגמה שנקבעה עבור התוצאות של וורסיה N-1:

מגמה נוכחית (וורסיה N)		המגמה שנקבעה בוורסיה N-1
ירידה של פחות מ-2% ביחס ל-N-1	עלייה של פחות מ-2% ביחס ל-N-1	
ללא שינוי	עלייה	עלייה
ירידה	ללא שינוי	ירידה
ללא שינוי	ללא שינוי	ללא שינוי

הגישה הבסיסית שהטבלה מממשת:

Feature	Score
Power consumption	97 (mW scale: 90-98)
Transactions/mSec	66 (Trans./mSec scale: 20-100)
CPU utilization	24 (%) (scale: 5-15)
Memory usage	44 (MByte scale: 40-60)

שימו לב שאני עדיין משתמש בצבעים אבל הצביעה של קו התוצאות מסבירה את בחירת הצבע. בנוסף, היא מסבירה האם תוצאה גבוהה היא המועדפת או שמא התוצאה נמוכה. הצופה רואה בעצמו מה המשמעות, והצבע משמש בעיקר להפנות את תשומת הלב למה שאינו ירוק. כמובן שיצירה של דו"ח כזה לוקחת קצת זמן, אבל ודאי אפשר לאטמט את זה. אני בטוח שבמעבדה בקופת חולים לא יושב מישהו ומעדכן את תוצאות בדיקות הדם באופן ידני...

בנוסף לדיווח התוצאה, יש עוד שתי נקודות שצריך לפתור:

- איך התוצאה שקיבלנו בוורסיה הנוכחית משתווה לתוצאות קודמות? האם אנחנו במסלול שיפור או מתדרדרים? במילים אחרות מה המגמה (trend)?
- בדיקת ביצועים מחייבת לעיתים הכנת כמויות אדירות של נתונים (למשל בבדיקת ביצועים של בסיס נתונים; או דיוק של זיהוי עצמים מתוך תמונות) יתכן שבתחילת הפרויקט אין לנו עדיין מספיק נתונים, ומצד שני כבר התחלנו להריץ בדיקות ביצועים. איך לדווח תוצאות מוקדמות של בדיקות ביצועים, ובמקביל להעביר את המסר שזה תוצאות ראשוניות, מבוססות על נתונים חלקיים, ויתכן שישתנו משמעותית בעתיד?

חישוב המגמה

נניח שאנחנו מודדים את העומס על המעבד (CPU utilization). ירידה בעומס תצביע על שיפור ביעילות המוצר. אם התוצאה שמדדנו על הגרסה הנוכחית טיפה נמוכה מהערך שמדדנו על הגרסה הקודמת, האם נוכל לומר בוודאות שיש שיפור בגרסה החדשה? מסתבר שלא תמיד.

הרצת בדיקת העומס פעמים רבות על אותה וורסיה של התוכנה ועל אותה חומרה, תיתן פיזור מסוים של התוצאה. הסיבה לכך היא שבמקביל לקוד שלנו רצות במערכת עוד תוכנות רבות: שירותי מערכת ההפעלה, אנטי וירוס, כרטיס תקשורת שמגיב למסרים על הרשת, וכו' וכו' (בזמן שאני כותב שורות אלה רצות ברקע על המחשב שלי כ-150 תוכנות ושירותים). גם טמפרטורת הסביבה, רמת הבהירות של המסך ומידת הטעינה של הבטרייה משפיעות קצת על הביצועים. כל זה גורם לשונות במערכת הכוללת, ומכאן לשונות במדידות של ביצועי המעבד. בהרבה מקרים מכנים זאת "הרעש" שיש במדידה. כל זמן שהתוצאות של מדידה עדיין בתחום של הרעש, אי אפשר להגיד בוודאות שיש מגמה של שינוי.

אם נריץ את הבדיקה פעמים רבות, נוכל לחשב התפלגות של התוצאות ולקבל מושג על "היכולת של המערכת" (system capability). אם התוצאות מתפלגות לפי התפלגות מוכרת, אפשר לומר ברמה סבירה של בטחון מתי תוצאה אכן מראה על שינוי אמיתי. לדוגמה: אם ההתפלגות נורמלית (גרף פעמון) כל תוצאה שנופלת בתוך תחום של +/-3 סטיות תקן מהמוצע אינה בוודאות שינוי.

מתי כן נוכל לומר שיש שינוי? בכמה מקרים:

- אם המדידה חורגת מתחום ההתפלגות הרגיל (בהתפלגות נורמלית זה יהיה מעבר לתחום של 3- סטיות תקן מהמוצע)
- אם המדידה עדיין בתחום ההתפלגות, אבל מדידות על וורסיות חדשות מראות נטייה עקבית לאותו כיוון. בהתפלגות נורמלית, אם אין שינוי ביעילות הקוד, תוצאות של מדידות עוקבות יפלו



Feature	Result	Score	Trend
Power consumption	97%	98	→
Transactions/mSec	66 t/sec	100	→
CPU utilization	24%	15	↗
Memory usage	44 MB	60	→

במקרה הרע:	במקרה הטוב:
$\frac{(140 \cdot 58\% + 60 \cdot 75\%)}{200} = 63\%$	$\frac{(140 \cdot 93\% + 60 \cdot 75\%)}{200} = 87\%$

הדיווח אם כך יהיה:

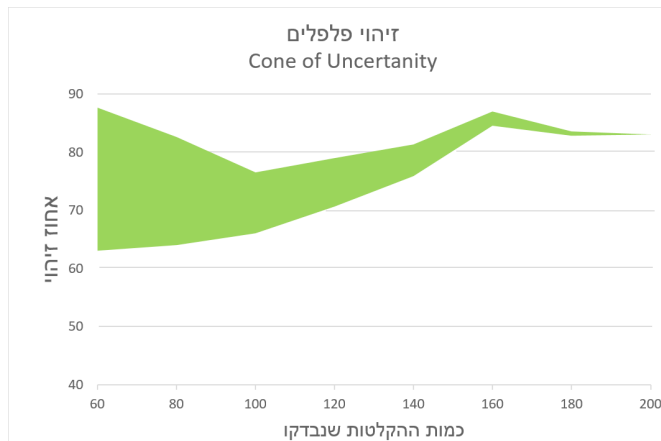
פרמטר	טווח תוצאה צפויה	כמות הבדיקות שבוצעו
זיהוי פלפלים פגומים	63% - 88%	30% [60 מתוך 200]

ככל שנתקדם בהקלטות ובהרצת הבדיקות, יהיו לנו יותר תוצאות אמיתיות ופחות אי וודאות לגבי התוצאה הצפויה בסוף. למשל, אם אחרי 120 הקלטות העשירון התחתון מראה זיהוי של 67%, העשירון העליון זיהוי של 88% והממוצע הוא 73%, הערכת הטווח תהיה:

במקרה הרע:	במקרה הטוב:
$\frac{(80 \cdot 67\% + 120 \cdot 73\%)}{200} = 70.6\%$	$\frac{(80 \cdot 88\% + 120 \cdot 73\%)}{200} = 79.0\%$

שימו לב שטווח אי הוודאות הצטמצם: 8.4% לעומת 24.5% אחרי 60 בדיקות.

ככל שנתקרב להרצה המלאה של כל הבדיקות המתוכננות, טווח אי הוודאות ירד. גרפית זה יראה משהו כזה (ועכשיו גם ברור למה זה נקרא "קונוס אי הוודאות" – הגרף נראה כמו קונוס ששולך וצר ככל שיש לנו יותר מידע):



גם אם זיהוי פלפלים הוא לא תחום העיסוק שלכם, אני מניח שיש מדידות שאתם עושים שהדיווח עליהם הוא ממוצע של בדיקות רבות. בכל המקרים האלה, הרעיון של קונוס אי וודאות יכול להיות רלוונטי ומאפשר דיווח חלקי, אך נכון, כבר בשלבים מוקדמים.

ועכשיו קצת Reality Check: חוץ מסימון המגמה, שאר הרעיונות שפרסמתי כאן הם עדיין בגדר רעיונות. כלומר: לא ראיתי מישהו שמימש אותם בפרויקט אמיתי. אני מפרסם אותו כאן בתקווה שמישהו מכם יוכל להשתמש בזה.

ספרו לי אם זה עזר לכם!

אם כיוון השינוי (הקטן) בוורסיה N ממשיך באותו כיוון של שינוי משמעותי שקרה בוורסיה N-1, הרי שהשינוי ממשיך להיחשב משמעותי. אם יש שבירת כיוון, אז השינוי אינו נחשב משמעותי. זה מימוש פשוטני ביותר ועקב כך אינו מושלם. למשל, הוא מאפשר לשינוי קטן אך מונוטוני לחמוק ללא השגחה (עליה במדידה בכל וורסיה, אבל בפחות מ-2% כל פעם). אפשר לשכלל את הגישה על ידי הסתכלות גם על התוצאות של וורסיה N-2, או חישובים של EWMA (ממוצע משוקלל נע).

https://en.wikipedia.org/wiki/EWMA_chart

אפשר ורצוי להוסיף את המגמה לתצוגה של התוצאות (בדוגמה זאת גם הוספתי עמודה עם התוצאות של המדידות – כהדגמה לווריאציה אפשרית על הרעיון הבסיסי):

דיווח על תוצאות ראשוניות

כאשר אתם בתחילת הפרויקט, יתכן וטרם מימשתם את כל הבדיקות שעל פיהם תקבעו את ביצועי המערכת. אתם כבר מריצים קצת בדיקות ביצועים ומעוניינים לפרסם את המידע שיש לכם אבל אתם חוששים. ידוע לכם שהתוצאות הן ראשוניות, וכשכל הבדיקות יפותחו ויורצו יתכן שהתוצאה תהיה שונה באופן משמעותי מהתוצאות הראשוניות. איך אפשר, מצד אחד לדווח, ומצד שני להעביר מסר ברור שהתוצאות ראשוניות ויתכן שישתנו בהמשך באופן קיצוני? רק להגיד את זה לא תמיד עוזר. אנשים יצטוו את המספר שדיווחתם – לא את המלל וההסתייגויות שליוו אותו.

ניקח דוגמה: אתם מפתחים מערכת אוטומציה לאריזה של פלפלים. המערכת משתמשת במצלמה ובאפליקציית עיבוד תמונה על מנת לזהות פלפלים פגומים ולשלוף אותם מפס האריזה. על מנת לבדוק את המערכת

אתם מתכננים להקליט את הווידאו הנקלט במצלמה כאשר על הפס עוברים פלפלים טובים מעורבים בפלפלים עם פגיעות שונות. סרטים אלה יזונו במעבדה לאלגוריתם שמהה פלפלים פגומים ואתם תבדקו כמה מהפגומים זווהו, וכמה מהפלפלים הטובים זווהו בטעות כפגומים. הקלטות אלה לוקחות זמן רב, ועד עכשיו הקלטתם רק 60 מתוך 200 ההקלטות המתוכננות. הרצתם את ההקלטות דרך המערכת, וקיבלתם שרמת הזיהוי של הפלפלים הפגומים היא 75%. לדווח או לא? הרי יכול להיות ששאר ההקלטות יורו על תוצאות טובות יותר, ואז סתם יצרתם פאניקה. מצד שני יכול להיות שהתוצאות על שאר ההקלטות יהיו גרועות, ואז יצרנו הרגשת בטחון שאינה במקומה והפתעה לא נעימה בהמשך.

הפתרון הוא לדווח, אבל בצורה שתעביר מסר של אי וודאות. אני עושה כאן שימוש של רעיון שנקרא "Cone of Uncertainty" – https://en.wikipedia.org/wiki/Cone_of_Uncertainty

בתור התחלה, נסתכל על התפלגות התוצאות שקיבלנו עבור 60 ההקלטות שכבר הרצנו. לצורך הדוגמה נאמר שהסתבר לנו שב-10% מההקלטות, זוהו 58% או פחות מהפלפלים הפגומים. ואילו ב-10% מההקלטות זוהו 93% או יותר מהפגומים. כלומר שטווח הזיהוי בהקלטות שבין העשירון התחתון לעליון הוא זיהוי של 58% - 93%. נעשה עכשיו הנחה שהתפלגות זו מגדירה פחות או יותר את יכולת המערכת שלנו, ושבהקלטות הבאות אלה יהיו התוצאות הקיצוניות האפשריות.

- במקרה הטוב ביותר, כל 140 ההקלטות הבאות יראו זיהוי של 93%

- במקרה הרע ביותר, כל ההקלטות יראו זיהוי של 58%

מה אם כך התוצאה האפשרית הסופית, בהתבסס על הסטטיסטיקה שבידנו כרגע, ובידיעה שיהיו עוד 140 הקלטות?

