

# Výber modelu testovania aplikácií dátových skladov

## Choice of datawarehouse applications testing model

*Andrej Trnka, Katedra informatiky FPV UCM v Trnave*

**Abstract:** Article describes choice of datawarehouse testing model. It compares two development life cycle models which are oriented to different types of testing. Next it describes differences between static and dynamic testing of datawarehouse applications.

**Key words:** datawarehouse, testing, V-model, W-model.

**Abstrakt:** Článok popisuje výber modelu testovania aplikácií dátových skladov. Porovnáva dva modely životného cyklu, ktoré sú zamerané na rozličné typy testovania. Ďalej popisuje rozdiely medzi statickým a dynamickým testovaním aplikácií dátových skladov.

**Kľúčové slová:** dátový sklad, testovanie, V-model, W-model.

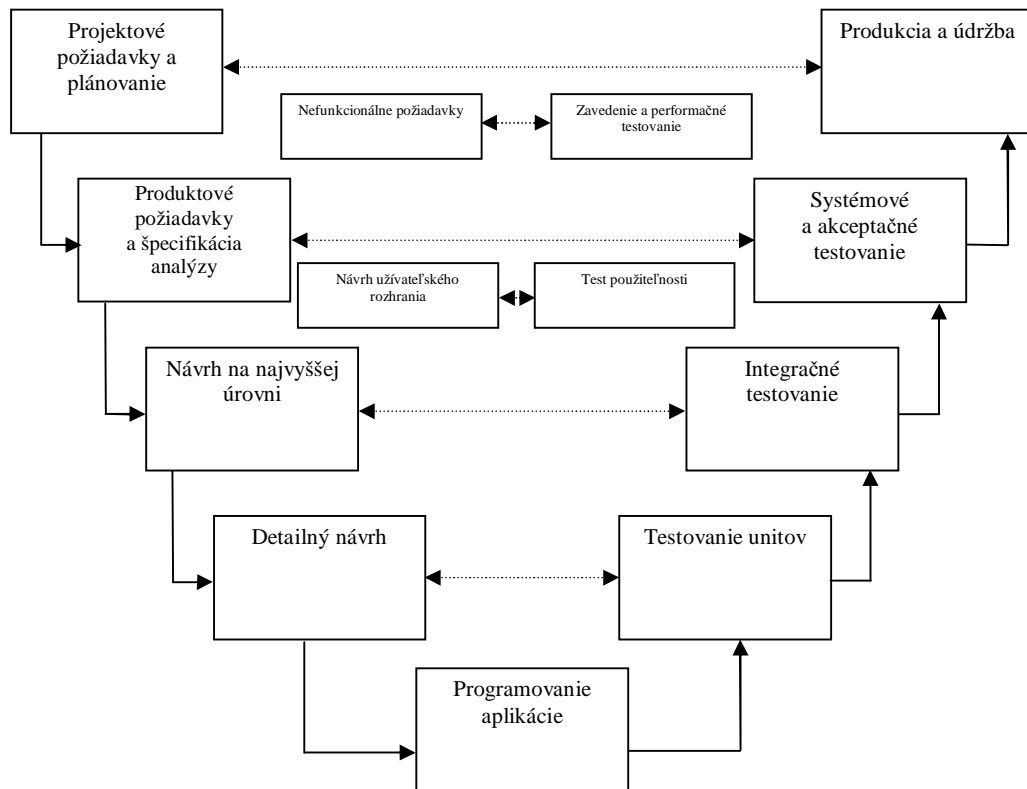
### 1. Úvod

Z pohľadu testovania aplikácií dátových skladov majú viaceré modely vývoja rôzne nedostatky:

- Testovacie aktivity začnú až po implementácii.
- Spojenie medzi rôznymi testovacími stupňami nie je veľmi jasné.
- Tiež nie je jasné spojenie medzi testovaním, odlaďovaním a zmenou úloh počas testovacej fáze.

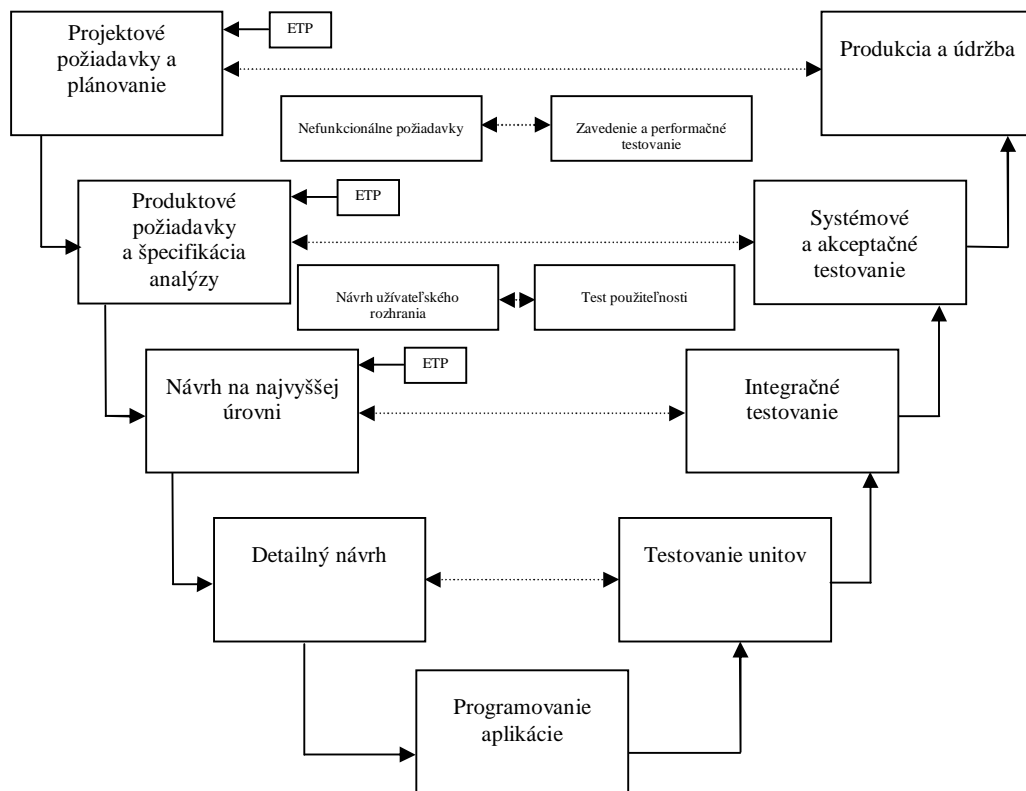
### 2. V-model

Testovanie aplikácií dátových skladov je možné popísať tzv. V-modelom. Tento popisuje myšlienku dynamického testovania (pravá strana modelu) použitím dokumentácie identifikovanej na ľavej strane modelu, ako základ pre testovanie. V-model ďalej popisuje predstavu skoršej prípravy testovania (Early Test Preparation - ETP).



**Obrázok 1: V-model**

Pomocou ETP sa dajú nájsť chyby v základných líniách, čo je efektívny spôsob včasného odhalenia chýb. Tento prístup je v princípe v poriadku a ETP je vždy účinné. Aj napriek tomu však existujú problémy s bežne prezentovaným V-modelom.



**Obrázok 2: V-model s implementáciou ETP**

Vzťah medzi dokumentmi na ľavej strane modelu a testovacími aktivitami na pravej strane nie je vždy dokonalý. Napr. funkcionálne špecifikácie obvykle neposkytujú dostatočné informácie pre systémové testovanie. Systémové testovanie musí často počítať s rôznymi aspektmi požiadaviek od užívateľa a rovnako musí počítať aj napr. s fyzickými problémami návrhu. Systémové testovanie k dôkladnému plánovaniu obvykle používa niekoľko zdrojov požadovaných informácií.(1)

### 3. Výhody a nevýhody V-modelu

V-model má niekoľko dobrých vlastností:

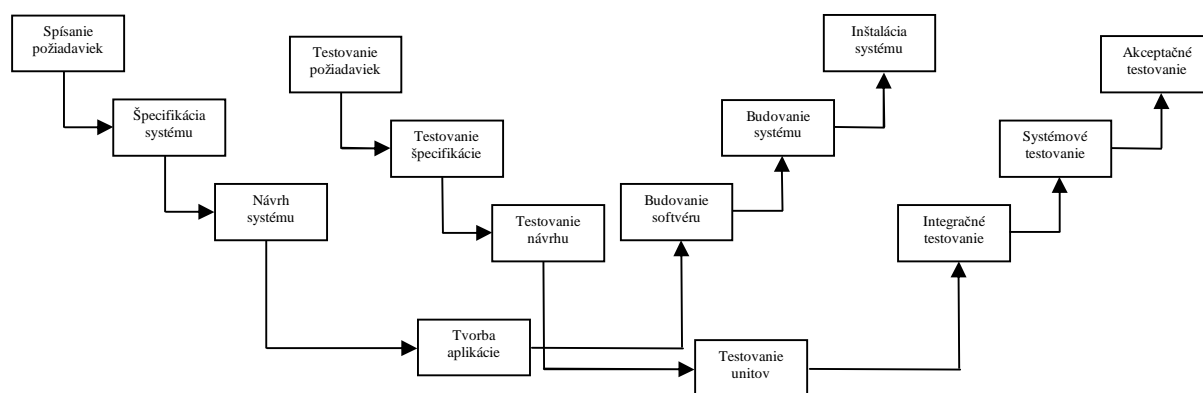
- Definuje hmatateľné fáze procesu a navrhuje logickú sekvenciu prístupnosti k jednotlivým fázam.
- Definuje logické vzťahy medzi jednotlivými fázami.
- Požaduje, aby testovacia dokumentácia bola čo najskôr. Napr. dokumentácia k integračnému testovaniu sa píše vtedy, keď jedokončený návrh na najvyššej úrovni a dokumentácia k testovaniu unitov sa píše vtedy, keď je dokončený detailný návrh.
- Poskytuje rovnakú váhu vývoja a testovania.

- Poskytuje jednoduchý postup procesu vývoja softvéru.

Vo všetkých svojich častiach však V-model hovorí málo o statickom testovaní. V-model zachádza s testovaním ako s back-door aktivitou na pravej strane modelu. V modeli nie je žiadna zmienka o potencionálne väčšej hodnote statického testovania ako je napr. statická analýza kódu. Toto je najvýznamnejšie opomenutie a V-model nepodporuje širší pohľad testovania.(3)

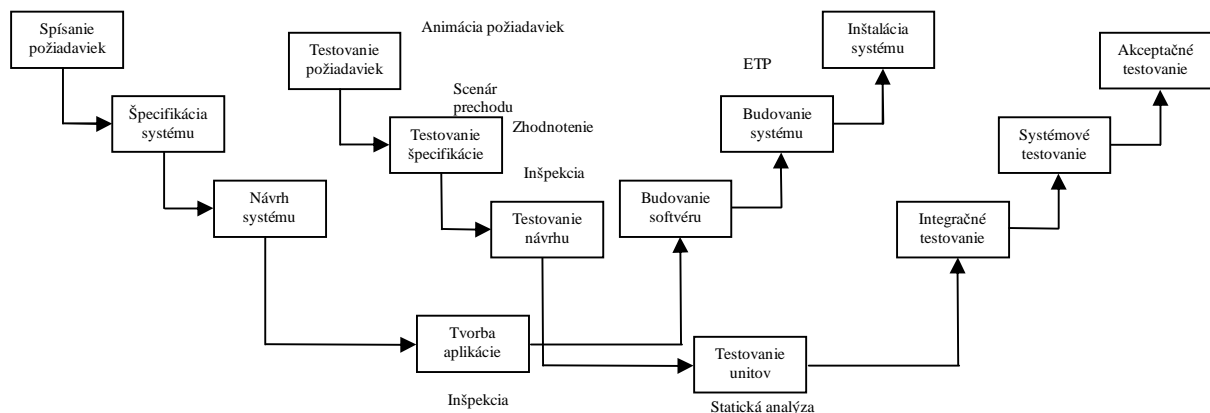
#### 4. Odstránenie nedostatkov V-modelu

Nedostatky V-modelu odstraňuje tzv. W-model. Návrh tohto modelu bol predstavený už v roku 1993. W-model sa zaostruje na vývoj produktov na rozdiel od V-modelu, ktorý sa zaostroval na dynamiku testovania.



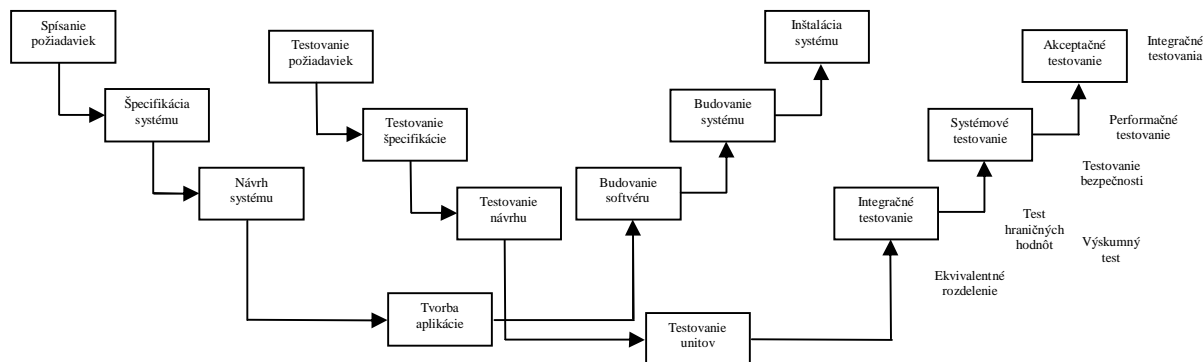
**Obrázok 3: W-model**

V podstate každá vývojová aktivita, ktorá niečo produkuje, je zatienená testovacou aktivitou. Účelom testovacej aktivity je určiť, či sa stretli ciele vývojovej aktivity s požiadavkami užívateľa. W-model predstavuje štandardný vývojový cyklus každého stupňa vývoja odzrkadlený testovacou aktivitou. Na ľavej strane sú vývojové aktivity doprevádzané testovacími aktivitami. Aj keď má organizácia rozličnú sadu vývojových stupňov, W-model je veľmi ľahko prispôsobiteľný každej situácii. Najdôležitejšou myšlienkou je to, že W-model sa zameriava na riziká v bode, kde môže byť testovanie najefektívnejšie.



**Obrázok 4: W-model a statické techniky testovania**

Pri zameraní sa na techniky statického testovania je na ľavej strane W-modelu vidieť široký okruh techník dostupných pre evaluáciu produktov.



**Obrázok 5: W-model a dynamické techniky testovania**

Pri uvažovaní o technikách dynamického testovania je vidieť široký okruh techník dostupných pre evaluáciu softvéru a systémov. Testovanie unitov, integračné testovanie, systémové testovanie a akceptačné testovanie sa môžu použiť v návrhu funkcionálneho testovania práve tak ako nefunkcionálne techniky testovania.

W-model odstraňuje dosť umelé obmedzenia, ktorých počet je rovnaký pri dynamickom testovaní a vývoji. Ak je pri návrhu a tvorbe aplikácie napr. päť stupňov vývoja, pri dynamickom testovaní môžu byť potrebné len tri stupne testovania. Systémové a akceptačné testovanie môže byť vhodné pri bežnom spôsobe práce. Teda testovanie celého projektu by mohlo byť robené len na troch stupňoch a nie na piatich. W-model však nepresadí projektovú súmernosť, ktorá v reáli neexistuje (alebo nemôže existovať). W-model nezavádza žiadne pravidlo, ktoré by hovorilo o tom, že dynamické testy musia byť založené na dokumentoch v špecifických stupňoch (aj keď produktová dokumentácia sa skoro vždy používa ako základný prvok pre dynamické testovanie).

Používaniu W-modelu musí však predchádzať:

- Identifikácia špecifických rizík.
- Výber testovacích techník.
- Plánovanie testovacích aktivít.(3)(1)

## 5. Rozdiely medzi statickým a dynamickým testovaním

- Statické testovanie je o prevencii a dynamické o odstraňovaní chýb.
- Nástroje statického testovania poskytujú väčšie benefity.
- Statické testovanie je veľakrát cenovo výhodnejšie než dynamické.
- Statické testovanie je efektívnejšie.
- Statické testovanie poskytuje komplexnú diagnostiku kódu.
- Dynamické testovanie obvykle trvá dlhšie než statické.
- Dynamické testovanie nachádza menej chýb než statické.
- Statické testovanie môže byť robené pred kompiláciou, zatiaľ čo dynamické testovanie sa môže vykonať len po kompilácii aplikácie.
- Statické testovanie môže nájsť syntaktické chyby, ťažko udržiavateľný kód, kód neprispôsobený štandardu atď. Toto nie je možné odhaliť pri dynamickom testovaní.(4)

## 6. Záver

Statické testovanie je nemenej dôležité ako testovanie dynamické. Z predchádzajúceho porovnania je vidieť, že V-model úplne ignoroval statické testovanie. Na rozdiel od neho, W-model pracuje nielen so statickým testovaním, ale aj s dynamickým testovaním. Oba tieto spôsoby sú pre testovanie aplikácií dátových skladov výhodnejšie.

Tento príspevok bol podporovaný grantovou agentúrou VEGA v rámci projektu číslo 1/4078/07, za čo autor vyslovuje poďakovanie.

## 7. Zoznam bibliografických odkazov

- (1) The W-Model, dostupné [on-line] na <http://www.gerrardconsulting.com>, [cit.: 3.3.2008]
- (2) LIVERSIDGE, Ed: The Death of V-Model, dostupné [on-line] na <http://www.harmonicss.co.uk>, [cit.: 3.3.2008]
- (3) Testing Models, dostupné [on-line] na <http://geekswithblogs.net>, [cit.: 3.3.2008]
- (4) What is the difference between static and dynamic testing?, dostupné [on-line] na <http://www.robdavispe.com>, [cit.: 3.3.2008]

## 8. Adresa autora:

Andrej Trnka, Ing.  
Katedra informatiky FPV UCM v Trnave  
Nám. J. Herdu 2  
917 01 Trnava  
[andrej.trnka@ucm.sk](mailto:andrej.trnka@ucm.sk)