

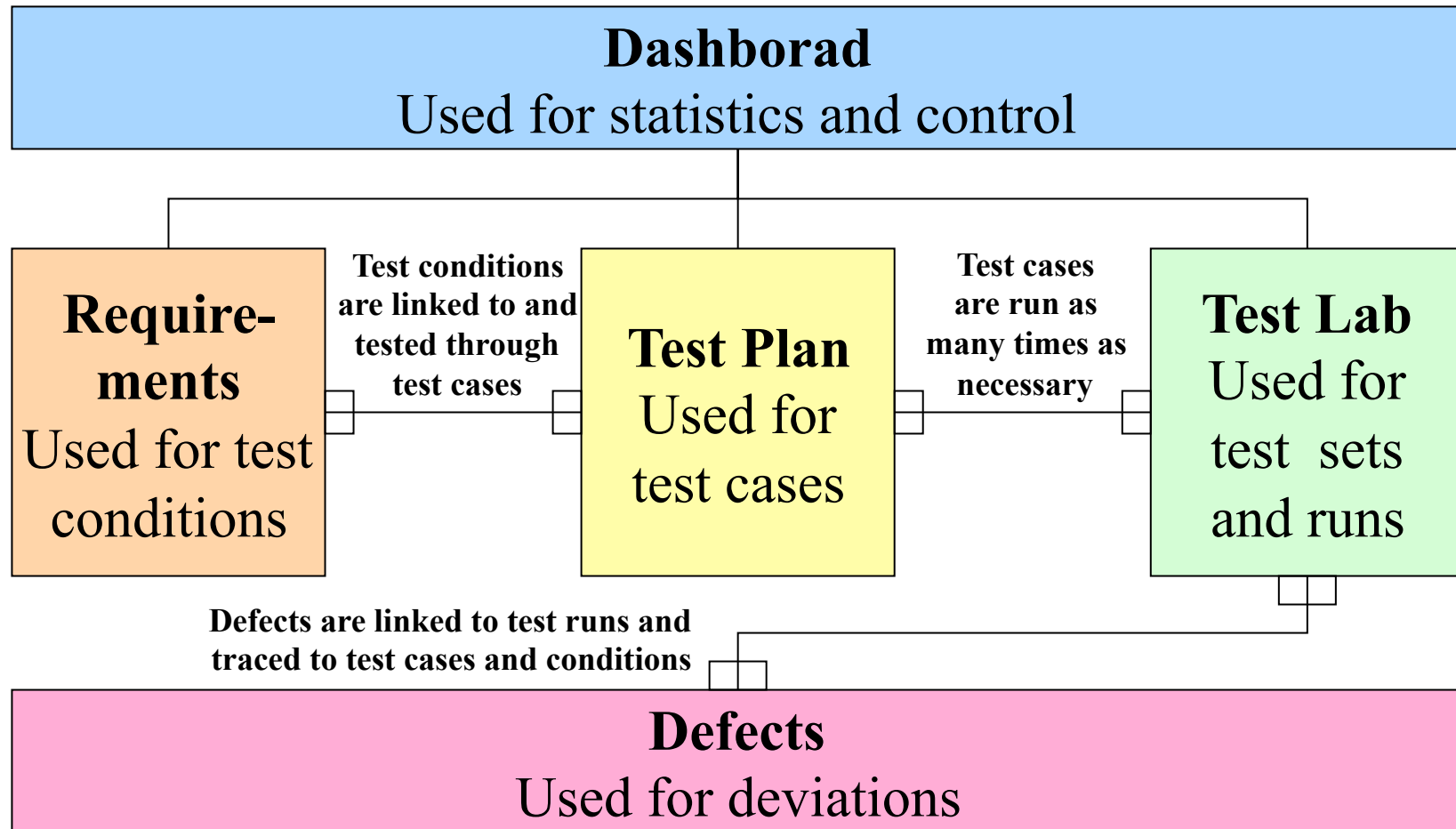
# Quality Center

Guide on how to use the test tool

# Why do we need a test tool when we already know the solution?

- To list all steps in the flow that need to be verified ("test conditions")
- To list all activities to execute the flow ("test cases")
- To document test progress ("test runs")
- To document and follow up deviations ("defects")
- To obtain statistics on how well tested the solution is (Dashboard)

# How is Quality Center organized?



# How do I log in to Quality Center?

1. Navigate to [http://w021/qcbin/start\\_a.htm](http://w021/qcbin/start_a.htm)
2. Authenticate with your user name and password
3. Login to project [Project Name]



The image shows the Quality Center login interface. It features a dark blue header with the text "Quality Center" on the left and a photograph of a person on a wooden deck on the right. Below the header is a light gray login form. The form contains the following elements:

- Login Name:** A text input field containing "knhj01".
- Password:** A text input field.
- Automatically log in to my last domain and project on this machine
- Authenticate:** A button.
- [Forgot Password](#): A link.
- Domain:** A dropdown menu showing "FOLKSAM".
- Project:** A dropdown menu showing "Solvens\_II".
- Login:** A button.

# Add/update test conditions in Requirements (optional)

**New folder**

**New test condition**

**Delete test condition**

**Folder structure**

- 04 Systemtest ETL
- 05 Systemtest Calculation
- 06 Systemtest Report
- 07 Acceptance Criteria

**Typical test condition structure**

- 01 Job: Does the job run?
- 02 Source: Where does data come from?
- 03 Target: Where does data flow to?
- 04 Calculations: What is calculated?

01	Job: Does the job run?	-----	15	knvj01
02	Source: Where does data come from?	-----	69	knhj01
03	Target: Where does data flow to?	-----	4692	knhj01
04	Calculations: What is calculated?	-----	2799	knhj01

# Add/update test cases in Test Plan (optional)

**New folder**

**New test case**

**Delete test case**

**Folder structure**

- 04 Systemtest ETL
- 05 Systemtest Calculation
- 06 Systemtest Report
- 07 Acceptance Criteria

**Typical test case structure**

- 01 Run job for SK and FFF
- 02 Verify input
- 03 Verify output
- 04 Verify calculations
- 05 Run negative tests

Step 2	Navigate to Products / SAS Folksam / Jobs /
Step 3	Run the job LOAD_MAP_BUSINESS_ACTIVITY_
Step 4	Start SAS Enterprise Guide and log in with your
Step 5	Navigate to Servers / SASApp / Libraries / RRR
Step 6	Review the indata table columns listed in the foll

# Prepare test sets in Test Lab

The screenshot displays the Test Lab interface with several callout boxes highlighting key features:

- New folder**: Points to the folder creation icon in the toolbar.
- New test run**: Points to the test run icon in the toolbar.
- Delete test run**: Points to the delete icon in the toolbar.
- Add tests to run**: Points to the 'Add tests to run' button in the toolbar.
- Folder structure**: A callout box listing the following structure:
  - 04 Systemtest ETL
  - 05 Systemtest Calculation
  - 06 Systemtest Report
  - 07 Acceptance Criteria
- Test cases in run**: Points to the '01 Run Jobs' folder in the tree view.

The main interface shows a tree view on the left with the following structure:

- Root
  - Unattached
  - PoC Solvens II
  - Solvens II
    - Solvens II - Rapporteringsprojekt utdata
      - 01 Iteration 1
        - 01 Sprint 1
        - 02 Sprint 2
        - 03 Sprint 3
        - 04 Sprint 4
        - 05 Systemtest
          - 04 Systemtest ETL
            - 01 Run Jobs
          - 05 Systemtest Beräkning
          - 06 Systemtest Rapport
        - A Leverabler

The right pane shows a table of test cases with columns for ID, Name, and Status:

ID	Name	Status
[1]09	Run PP_LOAD_PARAMETERS	MANUAL
[1]06	Run PP_LOAD_CONFIGURATION_DIM_LOOP	MANUAL
[1]13	Run PP_LOAD_MAP_BUSINESS_ACTIVITY_TYPE	MANUAL
[1]21	Run LOAD_DATE_REFERENCE_LOOP	MANUAL
[1]22	Run STG_PROJECT_OPTION_SET	MANUAL
[1]23	Run LOAD_PROJECT_OPTION_SET_LOOP	MANUAL
[1]24	Run STG_PROJECT_DIM	MANUAL
[1]25	Run LOAD_PROJECT_DIM_LOOP	MANUAL
[1]26	Run STG_X_CASHFLOW_DETAIL	MANUAL
[1]07	Run PP_LOAD_PARAMETERS	MANUAL
[1]08	Run PP_LOAD_CONFIGURATION_DIM_LOOP	MANUAL
[1]05	Run LOAD_MAP_CASHFLOW_TYPE	MANUAL
[1]04	Run LOAD_MAP_BUSINESS_ACTIVITY_TYPE	MANUAL

# Run test runs in Test Lab

1. Select test
2. Press Run
3. Mark each step  
Pass/Fail
4. Type result and  
create defect
5. Add/delete step  
if appropriate
6. End Run
7. Click Yes  
for updates

The screenshot shows the Test Lab interface with several callouts pointing to specific features:

- Add step**: Points to the plus icon in the toolbar.
- Delete step**: Points to the minus icon in the toolbar.
- Pass**: Points to the green checkmark icon in the toolbar.
- Fail**: Points to the red X icon in the toolbar.
- Create defect**: Points to the red flag icon in the toolbar.
- Attach calculation if appropriate**: Points to the paperclip icon in the toolbar.
- Use designs and update expected result if appropriate**: Points to the refresh icon in the toolbar.
- Type actual result**: Points to the 'Verkligt resultat' text area.

The main interface displays a table with the following data:

Steg	Status	Exec Date	
Step 6	No Run	2012-04-04	11:08:19
Step 7	No Run		
Step 8	No Run		
Step 9	No Run		

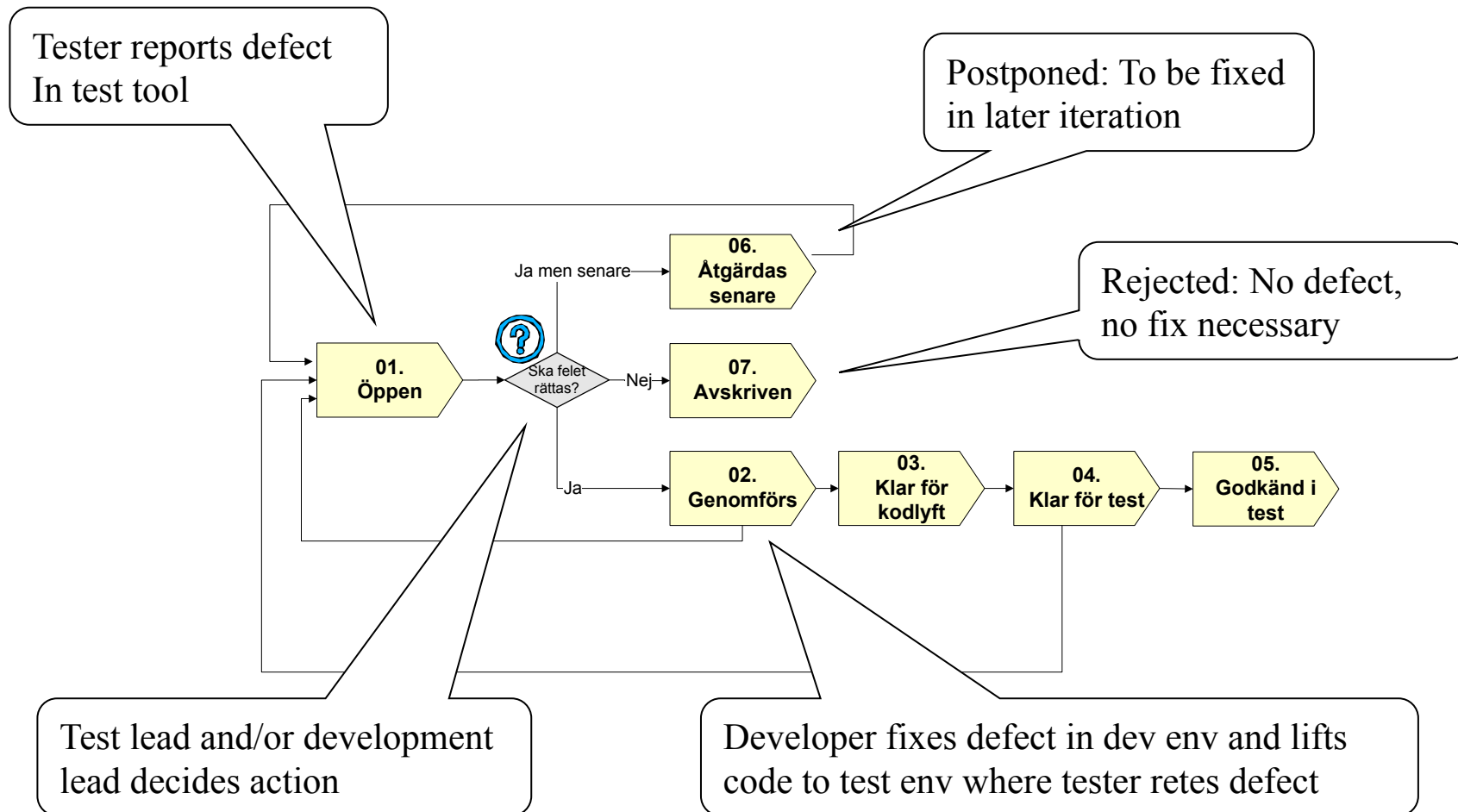
Below the table, there are two text areas for results:

**Förväntat resultat:**  
Distinct values have been written using replace load method

**Verkligt resultat:**

A 'Confirm' dialog box is open, asking: "Do you want to save the changes into test planning?" with 'Yes' and 'No' buttons.

# Follow Defect Process



# Create defects from Test Lab or directly in Defects

- **Sammanfattning:**  
Summary of defect
- **Tilldelad person:**  
Developer to investigate
- **Testområde:**  
Defect area (ETL, Calc, Report)
- **Feltyp:**  
Error, new requirement,  
unclear, cosmetic, stop
- **Priority:**  
Acute, high, middle, low
- **System:**  
Solvens II
- **Detected in Release:**  
Current release
- **Target Release:**  
Same release

**New Defect**

Clear Attach: [Icons]

\* Sammanfattning: [Text Input]

Details

* Status felrapport:	01.Öppen	* Feltyp:	Fel
* Tilldelad person:	[Dropdown]	Prioritet:	Medel
Detected in Cycle:	[Dropdown]	Detected in Release:	[Dropdown]
Testområde:	[Dropdown]	System:	[Dropdown]
Target Cycle:	[Dropdown]	Target Release:	[Dropdown]

Beskrivning: [Text Area]

Submit Close

# Update defects in Defects and retest test cases in Test Lab

1. Developer searches for assigned defects in field "Tilldelad person"
2. Developer documents measures, update defect and reassign to tester
3. Tester retests in Test Lab and approves the defect in Defects

## Update status

1. Open: Defect detected by tester
2. In progress: Defect assigned to developer
3. Ready for code lift: Fixed in dev env
4. Ready for test: Fixed in test env
5. Approved in test: Defect retested
6. Postponed: To be fixed in later iteration
7. Rejected: No defect, no fix necessary

## Update assignee

## Update target release

## Add comment

What was done to fix the defect  
OR why is it not a defect

The screenshot shows a web application interface for defect management. At the top, it displays 'Domain: FOLKSAM, Project: Solvens II, User: knhj01' and a 'Logout' button. Below this is a search bar and a table of defects. The table has columns for 'Status felrap', 'System', 'Target Cycle', 'Target Relea:', 'Testområde', and 'Tilldelad pers'. One defect is highlighted with a status of 'Godkänd i test' and assigned to 'knhj01'. A dropdown menu is open over the 'Godkänd i test' status, showing options from '01. Öppen' to '08. Löst på restlista'. A callout box titled 'Update status' lists seven status options. Another callout box titled 'Update assignee' points to the 'Tilldelad pers' column. A third callout box titled 'Update target release' points to the 'Target Relea:' column. A fourth callout box titled 'Add comment' points to the 'Beskrivning' field, which contains an error message: 'Line 497: ERROR: %SYSEVALF function has no expression to evaluate. Line 497: ERROR: Argument 1 to function PUTN referenced by the %SYSFUNC or %QSYSFUNC macro function is not a number.' The user 'Nicholas Hjelmsberg' is listed as the assignee for this defect.

Status felrap	System	Target Cycle	Target Relea:	Testområde	Tilldelad pers
Godkänd i test	Solvens II			Utdata ETL	knhj01

01. Öppen  
02. Genomförs  
03. Klar för kodlyft  
04. Klar för test  
05. Godkänd i test  
06. Åtgärdas senare  
07. Avskriven  
08. Löst på restlista

**Beskrivning** Attachments History

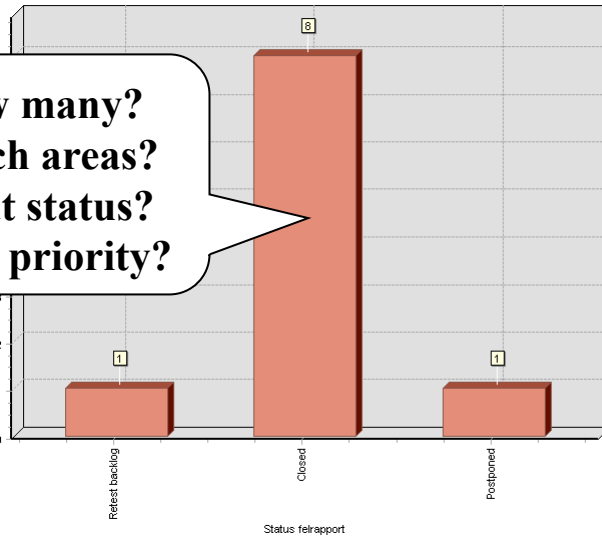
\* Sammanfattning: Insurance cashflow: Extract error %SYSEVALF

Beskrivning:  
**Verkligt resultat:**  
**RMFI Update load\_status table**  
Line 497: ERROR: %SYSEVALF function has no expression to evaluate.  
Line 497: ERROR: Argument 1 to function PUTN referenced by the %SYSFUNC or %QSYSFUNC macro function is not a number.

Nicholas Hjelmsberg  
EXTR\_INSURANCE\_CASHFLOW\_LOOP should be run.

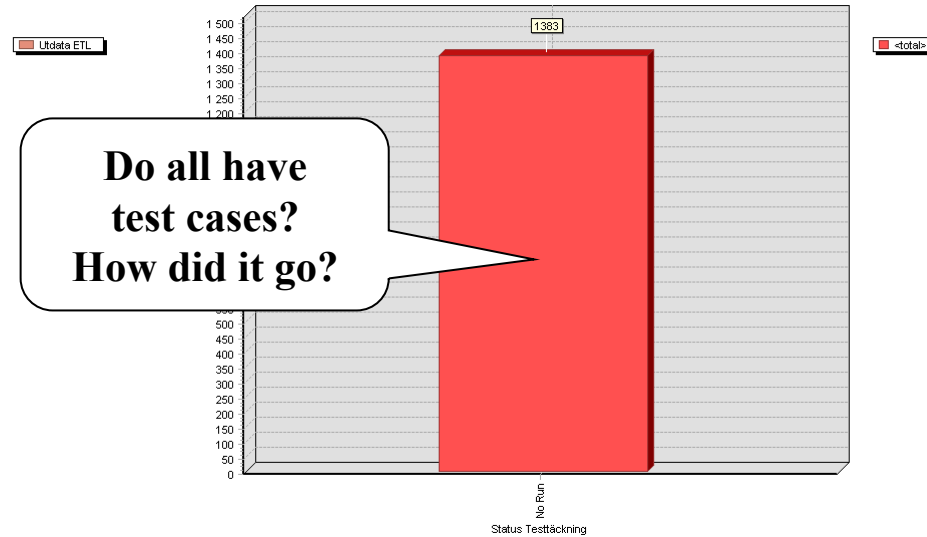
# View statistics in Dashboard

## Defects by Status/Priority



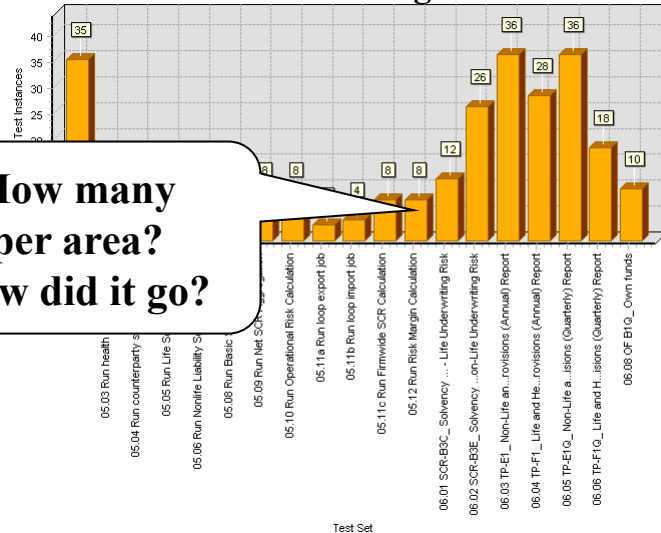
How many?  
Which areas?  
What status?  
What priority?

## Test Conditions Coverage



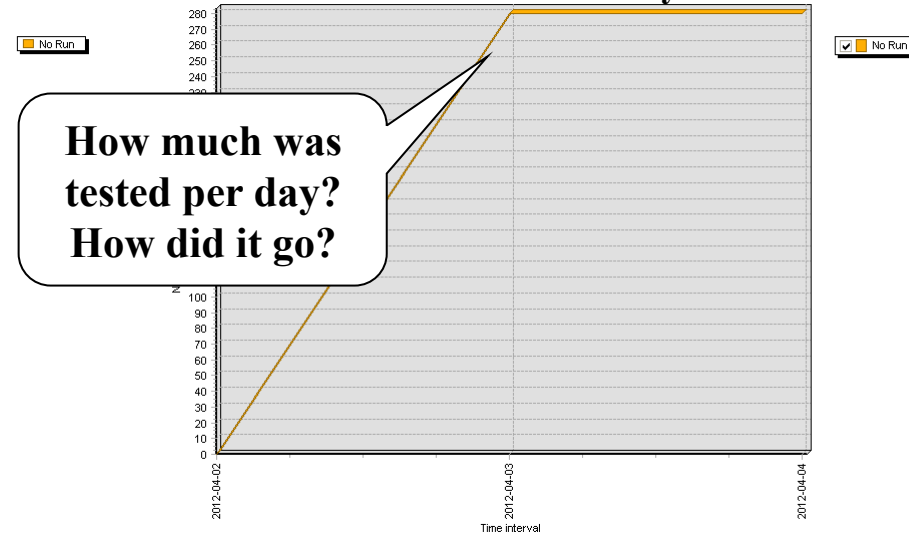
Do all have test cases?  
How did it go?

## Test Case Progress



How many per area?  
How did it go?

## Test Case Summary



How much was tested per day?  
How did it go?