ABAP Continuous Integration (CI) Plugin

Why automation is the key to code quality?

Andreas Gautsch

andreas.gautsch@gmx.at

@andreas_gautsch

Continuous Integration (CI) Certification Test *)

1. Continuous commits [activations] to one repository

2a. Automated checks of Build and Unit Tests after each commit

2b. Automated static code checks [ATC]

3. Unit Test Failures [and ATC errors] are repaired fast

*) adapted, see original idea and definition by Martin Fowler at:
https://martinfowler.com/bliki/ContinuousIntegrationCertification.html

The focus of the Plugin is on automation

Target: Automatic execution of existing ABAP in Eclipse features
after each activation of ABAP development objects

Purpose: Save development time and
Get fast feedback about the code state

ABAP CI Plugin

Target is primary the automatic visualisation of the actual state of the source code.
For solving unit test failures and cleaning up the particular ATC findings the standard ABAP in Eclipse views are much better and more powerful.

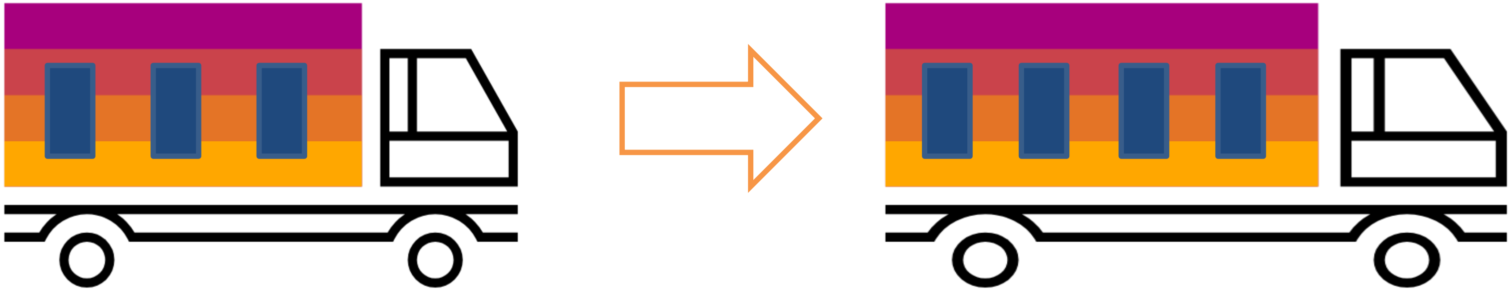
Automated Continuous Integration saves time

Area	Without CI	Manual CI tools	With abapCI Plugin
source code change	Equal		
Minimal mouse clicks or shortcuts necessary*)	2 1	**12** **4**	2 1
CI and quality components	None	AbapUnit ATC checks (Formatted code, Delete not used vars)	AbapUnit ATC checks (Formatted code Delete not used vars)
Project manager mood	😊	😐	😊

*) base for calculation: two changed ABAP classes

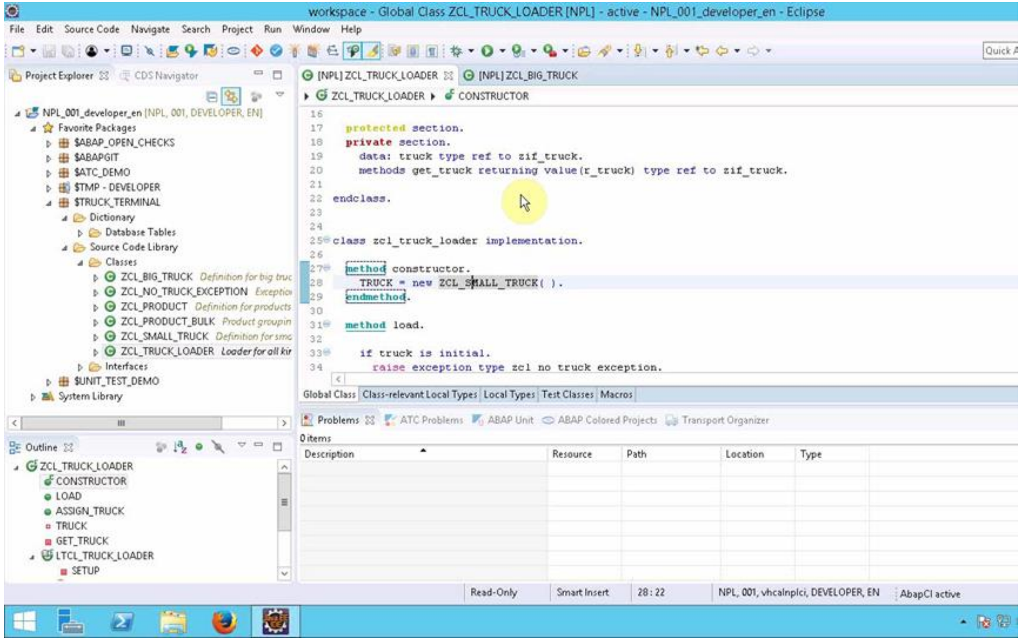
Coding example – Truck terminal

Use case: The maximal item **capacity can be increased from 3 to 4 items** as the new bought trucks have more loading space.



Images from https://www.sap.com

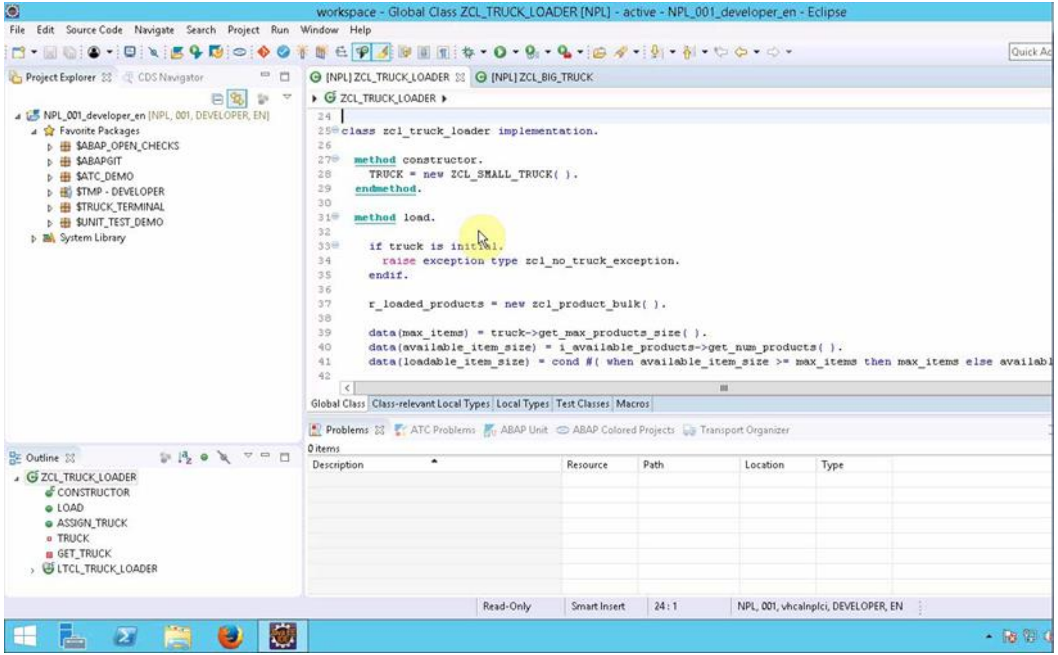
Skipping Continuous Integration is fast (… in the short time)



```
  protected section.
  private section.
    data: truck type ref to zif_truck.
    methods get_truck returning value(r_truck) type ref to zif_truck.

endclass.


class zcl_truck_loader implementation.

  method constructor.
    TRUCK = new ZCL_SMALL_TRUCK( ).
  endmethod.

  method load.

    if truck is initial.
      raise exception type zcl no truck exception.
```

YouTube - Comparison ABAP development Performance - Without CI tools

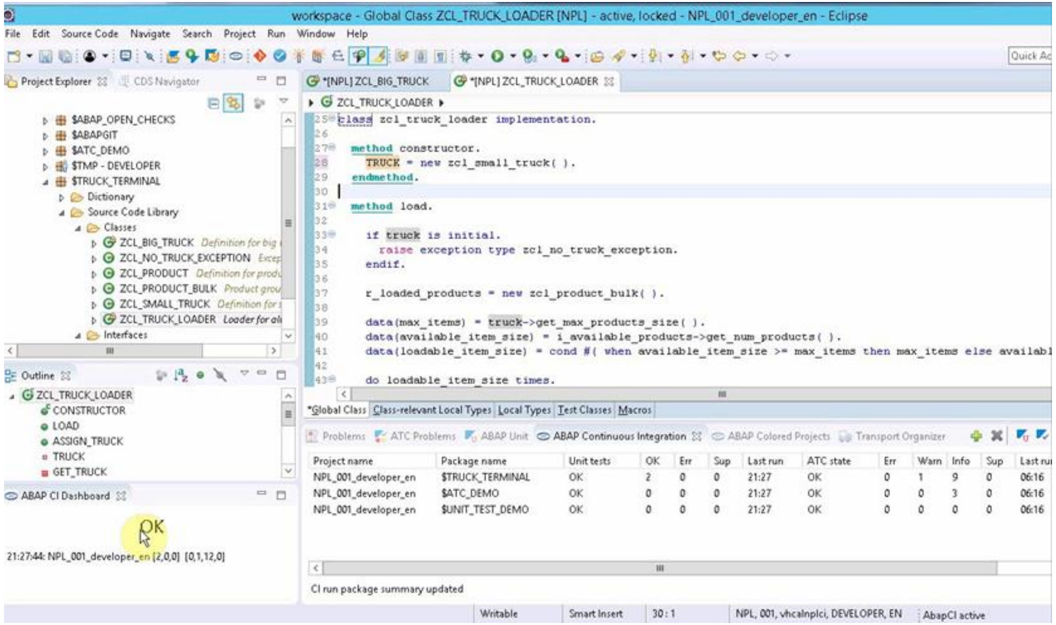
Manual usage of CI tools needs a lot of time



```
class zcl_truck_loader implementation.

  method constructor.
    TRUCK = new ZCL_SMALL_TRUCK( ).
  endmethod.

  method load.

    if truck is initial.
      raise exception type zcl_no_truck_exception.
    endif.

    r_loaded_products = new zcl_product_bulk( ).

    data(max_items) = truck->get_max_products_size( ).
    data(available_item_size) = i_available_products->get_num_products( ).
    data(loadable_item_size) = cond #( when available_item_size >= max_items then max_items else availabl
```

YouTube - Comparison ABAP development Performance - Manual CI tools

On purpose of ABAP CI plugin is to save time



```
class zcl_truck_loader implementation.

  method constructor.
    TRUCK = new zcl_small_truck( ).
  endmethod.

  method load.

    if truck is initial.
      raise exception type zcl_no_truck_exception.
    endif.

    r_loaded_products = new zcl_product_bulk( ).

    data(max_items) = truck->get_max_products_size( ).
    data(available_item_size) = i_available_products->get_num_products( ).
    data(loadable_item_size) = cond #( when available_item_size >= max_items then max_items else availabl

    do loadable_item_size times.
```

21:27:44: NPL_001_developer_en [2,0,0] [0,1,12,0]

Project name	Package name	Unit tests	OK	Err	Sup	Last run	ATC state	Err	Warn	Info	Sup	Last ru
NPL_001_developer_en	$TRUCK_TERMINAL	OK	2	0	0	21:27	OK	0	1	9	0	06:16
NPL_001_developer_en	$ATC_DEMO	OK	0	0	0	21:27	OK	0	0	3	0	06:16
NPL_001_developer_en	$UNIT_TEST_DEMO	OK	0	0	0	21:27	OK	0	0	0	0	06:16

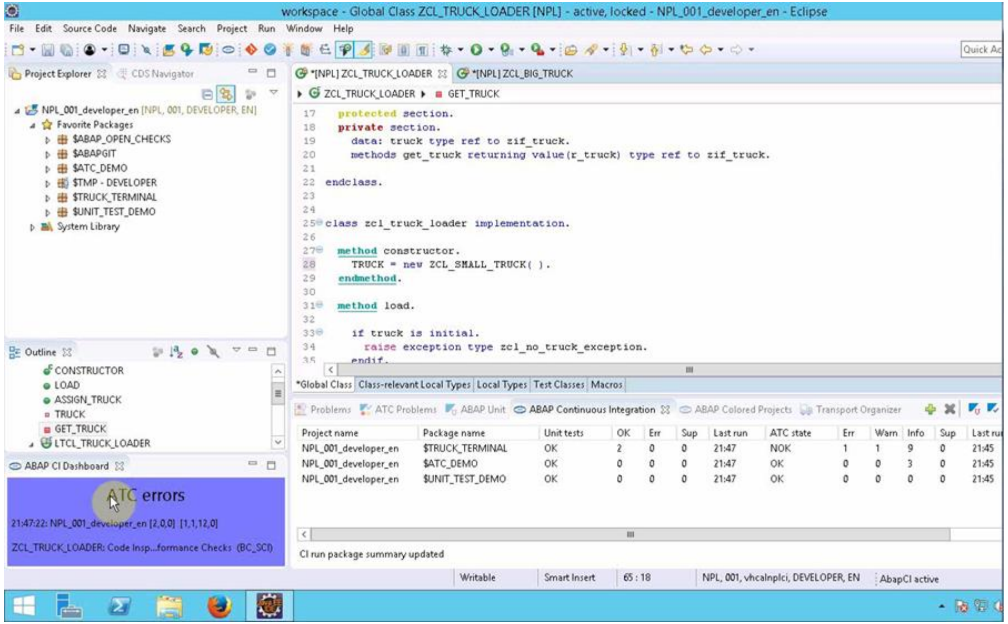
CI run package summary updated

YouTube - Comparison ABAP development Performance - With ABAP CI plugin

Using Continuous Integration prevents errors

Area	No code quality	Manual quality tools	With abapCi Plugin
source code change	Equal		
Minimal mouse clicks or shortcuts necessary*)	2 1	**12** **4**	2 1
CI and quality components	None	AbapUnit ATC checks (Formatted code, Delete not used vars)	AbapUnit ATC checks (Formatted code Delete not used vars)
Project Manager mood for 5 change requests	😊 😊 😊 😊 ☹	😐 😐 😐 😐 😐	😊 😊 😊 😊 😊

Automatic sourcecode state feedback



```
  protected section.
  private section.
    data: truck type ref to zif_truck.
    methods get_truck returning value(r_truck) type ref to zif_truck.

endclass.


class zcl_truck_loader implementation.

  method constructor.
    TRUCK = new ZCL_SMALL_TRUCK( ).
  endmethod.

  method load.

    if truck is initial.
      raise exception type zcl_no_truck_exception.
    endif.
```

Project name	Package name	Unit tests	OK	Err	Sup	Last run	ATC state	Err	Warn	Info	Sup	Last ru
NPL_001_developer_en	$TRUCK_TERMINAL	OK	2	0	0	21:47	NOK	1	1	9	0	21:45
NPL_001_developer_en	$ATC_DEMO	OK	0	0	0	21:47	OK	0	0	3	0	21:45
NPL_001_developer_en	$UNIT_TEST_DEMO	OK	0	0	0	21:47	OK	0	0	0	0	21:45

CI run package summary updated

ABAP CI Dashboard

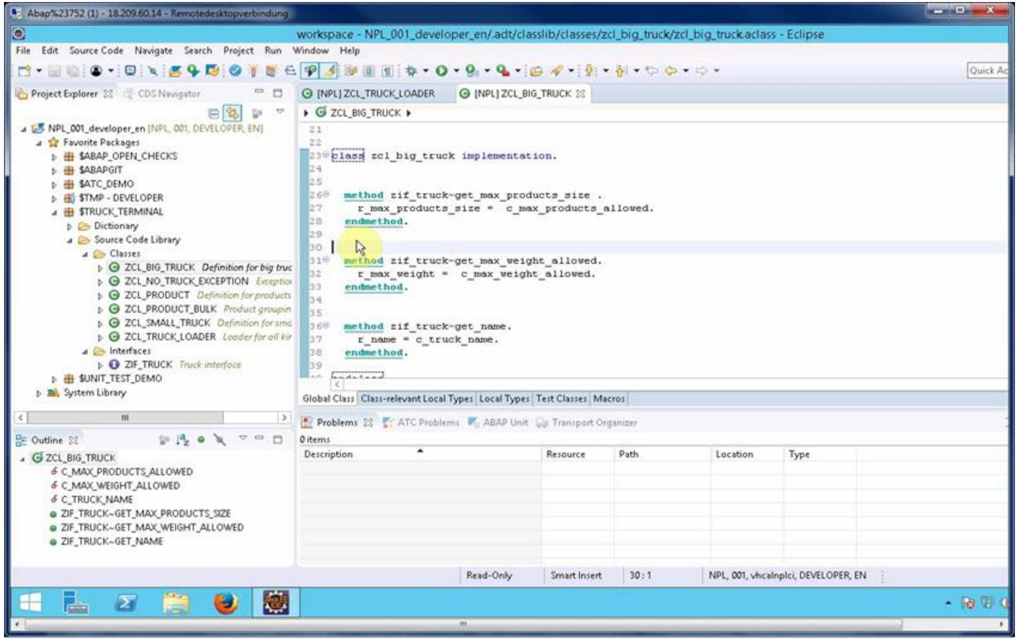
ATC errors

21:47:22: NPL_001_developer_en [2,0,0] [1,1,12,0]

ZCL_TRUCK_LOADER: Code Insp...formance Checks (BC_SCI)

YouTube - ABAP Continuous Integration Plugin - Sourcecode state feedback

Installing and configuring needs only 2 minutes


Abap%23752 (1) - 18.209.60.14 - Remotedesktopverbindung
workspace - NPL_001_developer_en/.adt/classlib/classes/zcl_big_truck/zcl_big_truck.aclass - Eclipse
File Edit Source Code Navigate Search Project Run Window Help
Project Explorer
CDS Navigator
[NPL] ZCL_TRUCK_LOADER
[NPL] ZCL_BIG_TRUCK
NPL_001_developer_en [NPL, 001, DEVELOPER, EN]
Favorite Packages
$ABAP_OPEN_CHECKS
$ABAPGIT
$ATC_DEMO
$TMP - DEVELOPER
$TRUCK_TERMINAL
Dictionary
Source Code Library
Classes
ZCL_BIG_TRUCK Definition for big truc
ZCL_NO_TRUCK_EXCEPTION Exceptio
ZCL_PRODUCT Definition for products
ZCL_PRODUCT_BULK Product groupin
ZCL_SMALL_TRUCK Definition for sma
ZCL_TRUCK_LOADER Loader for all kir
Interfaces
ZIF_TRUCK Truck interface
$UNIT_TEST_DEMO
System Library
Outline
ZCL_BIG_TRUCK
C_MAX_PRODUCTS_ALLOWED
C_MAX_WEIGHT_ALLOWED
C_TRUCK_NAME
ZIF_TRUCK~GET_MAX_PRODUCTS_SIZE
ZIF_TRUCK~GET_MAX_WEIGHT_ALLOWED
ZIF_TRUCK~GET_NAME
Global Class
Class-relevant Local Types
Local Types
Test Classes
Macros
Problems
ATC Problems
ABAP Unit
Transport Organizer
0 items
Description
Resource
Path
Location
Type
Read-Only
Smart Insert
30 : 1
NPL, 001, vhcalnplci, DEVELOPER, EN
Quick Ac

```
class zcl_big_truck implementation.


  method zif_truck~get_max_products_size .
    r_max_products_size =  c_max_products_allowed.
  endmethod.


  method zif_truck~get_max_weight_allowed.
    r_max_weight =  c_max_weight_allowed.
  endmethod.


  method zif_truck~get_name.
    r_name = c_truck_name.
  endmethod.

```

YouTube - ABAP Continuous Integration Plugin - Installation

Thanks for your attention!

Purpose: Save development time and
Give fast feedback

ABAP CI Plugin

Installation with: eclipse marketplace

Open Source
and documentation: GitHub

Question, Issues, Feature or Pull Requests welcome!

Appendix I

- Open Source Github repository for ABAP CI Plugin
  https://github.com/andau/abapCI

- Listing in Eclipse Marketplace
  https://marketplace.eclipse.org/content/abap-continuous-integration

- Continous Integration by Martin Fowler
  https://martinfowler.com/articles/continuousIntegration.html

- ATC – Blogs by , for example:
  https://blogs.sap.com/2017/02/27/remote-code-analysis-in-atc-for-developers/

Appendix II

- ABAP Test Cockpit (ATC) - Introduction Video on the ABAP channel on YouTube : https://youtu.be/_SmSJ8Tkl8w

- ABAP Open Checks by Lars Hvam https://github.com/larshp/abapOpenChecks