

The CT-Assist_Module 6.0

Add-On for the SAP® ABAP™ Workbench

A costs/benefit analysis

You can test the potential of the Add-On for yourself ...

Just take one of your recent programs and analyze it LIVE with the CT-Assist_Module.

This Add-On automatically types in your program again. As a special feature then marks those code parts which should have been inserted automatically by our CT-Assist_Module software assistants. Finally, a productivity statistic is generated.

The productivity statistic ...

Below we show you the statistic of a test program. This test program contains approximately **1000** lines of typical source code (no special variable names or statements were used).

Data	Description
57%	... of the ABAP™ program could have been automatically typed in with our code completion features (see line 6, statistics protocol). For example, you only have to type in mov and the CT-Assist_Module automatically suggests MOVE-CORRESPONDING , which you can then automatically insert into the code by pressing the TAB key.
4.424	characters were typed in by the user (see line 4, statistics protocol),
5.934	characters were automatically typed in by the CT-Assist_Module (see line 5, statistics protocol),
We now take a closer look at those 5.934 characters (which represent 57% of our ABAP™ program) that were entered by the code completion functions of the CT-Assist_Module:	
Automatically "typed in" ABAP™ statements (ABAP™ code completion)	
500	ABAP™ templates could have been inserted (see line 8, statistics protocol)
3.806	... these are 3,806 characters which were automatically typed in by our ABAP code completion function (see line 10, statistics protocol),
Automatically inserted variables (ABAP™ variable completion)	
229	variable names were recognized and inserted (see line 14, statistics protocol),
1.147	... these are 1,147 characters which were automatically typed in by our ABAP variable completion function (see line 15, statistics protocol)
Automatically inserted structures (ABAP™ structure completion)	
101	... structures were recognized (see line 17, statistics protocol) and were displayed with 6556 structure fields (see line 18, statistics protocol),
981	... these are 981 characters which were automatically typed in by our ABAP structure completion function (see line 19, statistics protocol)
5.934	The sum of the lines 10, 15, 19, statistics protocol is 5,934

The productivity statistics

Line	Title	Value	
1	Summary		
2	Number of program lines	980	Total number of program lines
3	Not analyzed lines	403	Comment lines and data declarations
4	Manual entered	4444	Chars manually entered by the user
5	Automatically entered	5961	Chars the user did not have to enter, because they were automatically entered by the CT-Module.
6	Percentage	57,28%	x% of the written source code was automatically entered by the CT-Module
7	Templates		
8	Inserted templates	501	How many times a command template was inserted ?
9	Chars until template	3	How many chars were statistically typed in before a template was inserted ?
10	Template chars	3823	Total chars that were automatically inserted by the template function
11	Variables		
12	Unique suggestions	138	How many times a unique variable tooltip suggestion was shown while typing ?
13	Multiple suggestions	178	How many times multiple variable suggestion were shown ?
14	Inserted suggestions	231	How many times these suggestion were taken by the user ?
15	Total inserted variable chars	1157	Total chars that were automatically inserted by the using the variable suggestion function
16	Structures		
17	Structures shown	101	How many times the elements of a structure (internal tables fields, DDIC fields, ...) were displayed ?
18	Shown structure fields	6556	How many structure elements were shown at all ?
19	Inserted chars	981	How many chars of a structure field were entered automatically ?
20	Information		
21	Mouse tooltip infos	0	How many times the user has pointed the mouse cursor on a variable and got further symbol/variable information ?
22	Symbols shown in info line	0	How many times further symbol/variable information was shown in the top symbol info line ?
23	Symbols shown in tree	0	How many times the corresponding symbol was automatically shown in the explorer tree ?
24	Other		
25	Lines of Code	996	Inserted lines via the return key.
26	String chars	1841	Chars that were entered within a string
27	Comment chars	0	Chars that were entered within a comment
28	Total chars	4444	Total manually entered chars
29	Command words	1283	Total inserted command and variable words
30	Commands	499	Total entered completed commands (which were terminated by a dot or comma)

A costs/benefit analysis

- **The time to write ABAP™ source code ("typing") is reduced dramatically**

According to our statistics **40 to 60%** of an ABAP™ program can be automatically entered by our code completion features. We estimate that you can save **120 minutes per day** compared to a **standard ABAP™ editor** (assuming, that work only 4-5 hours per day with the standard editor).

- **Time consuming search tasks (e.g. during the conception phase) are reduced**

When writing ABAP™ source code, the programmer often needs to know the exact names of certain variables, Data Dictionary structures (which are often not easy to remember, e.g. VBAK-KUNNR) and of course the exact ABAP™ statement names. Traditionally he/she had to look at many places to get this information (ABAP™ help, SAP® Data Dictionary, other ABAP™ include files, ...). This takes time.

The CT-Assist_Module provides much of this information right "at the fingertips" of the programmer. For example, pop up windows (automatically triggered by entering ABAP™ code) offer **Data Dictionary structures with their explanation. Declared variable names** are automatically shown and can then easily be inserted via mouse or TAB key. Features like the **Browser_Assist** help to query the complete source code of all related ABAP™ include members and show you all declared variables right away.

Even if you only spend 2 hours per day to search for this information, we estimate that you will be able to **save at least 60 minutes per day** (a conservative estimate in our opinion) by using the CT-Assist_Module.

- **The quality of ABAP™ coding is improved, ABAP™ programs will become more "readable"**

Just by using self-descriptive, **informative**, longer variable names the source code can become more readable and provide a better understanding of the program logic. This also means that training time for external or new programmers (who must understand the ABAP™ program first, before they can make changes) can be reduced. Thus the maintenance of the program is simplified.

By using the CT-Assist_Module, a programmer can use long variable names at no additional "typing" costs. He/she just has to enter only the first few letters of the variable name, and the rest is automatically inserted by the module.

Context sensitive help, live replace functions and the **logic group compression** (e.g. IF-ENDIF, FORM-ENDFORM ...) are further functions which will improve program handling and reduce the possibility of bugs during the design phase.

Obviously it is difficult to specify how much time could be saved by improving the quality of your ABAP™ code.

We estimate **nn minutes per day**. Please, decide for yourself how much time could be saved here; keep the lifetime of your software (maintenance) in mind.

- **Further time savings can be achieved by using the integrated ABAP™ command tree** (oe-step procedures)

Simply select (double-click, drag&drop, automatic code insert) ABAP™ statements from the interactive ABAP™ command tree (containing more than **3.500** entries). It is easy to choose ABAP™ commands AND their proper variants or additions. You will get syntax conformant code and this will reduce syntax errors in the development phase.

Our ABAP™ reference assist further suggests the most likely ABAP™ command as a tooltip. By pressing the TAB key, the command suggestion can be quickly inserted into the code.

More than **50** external developers have evaluated these functions and found them to be extremely useful.

We estimate that **15 -25 minutes per day** can be saved by using these features.

A result ...

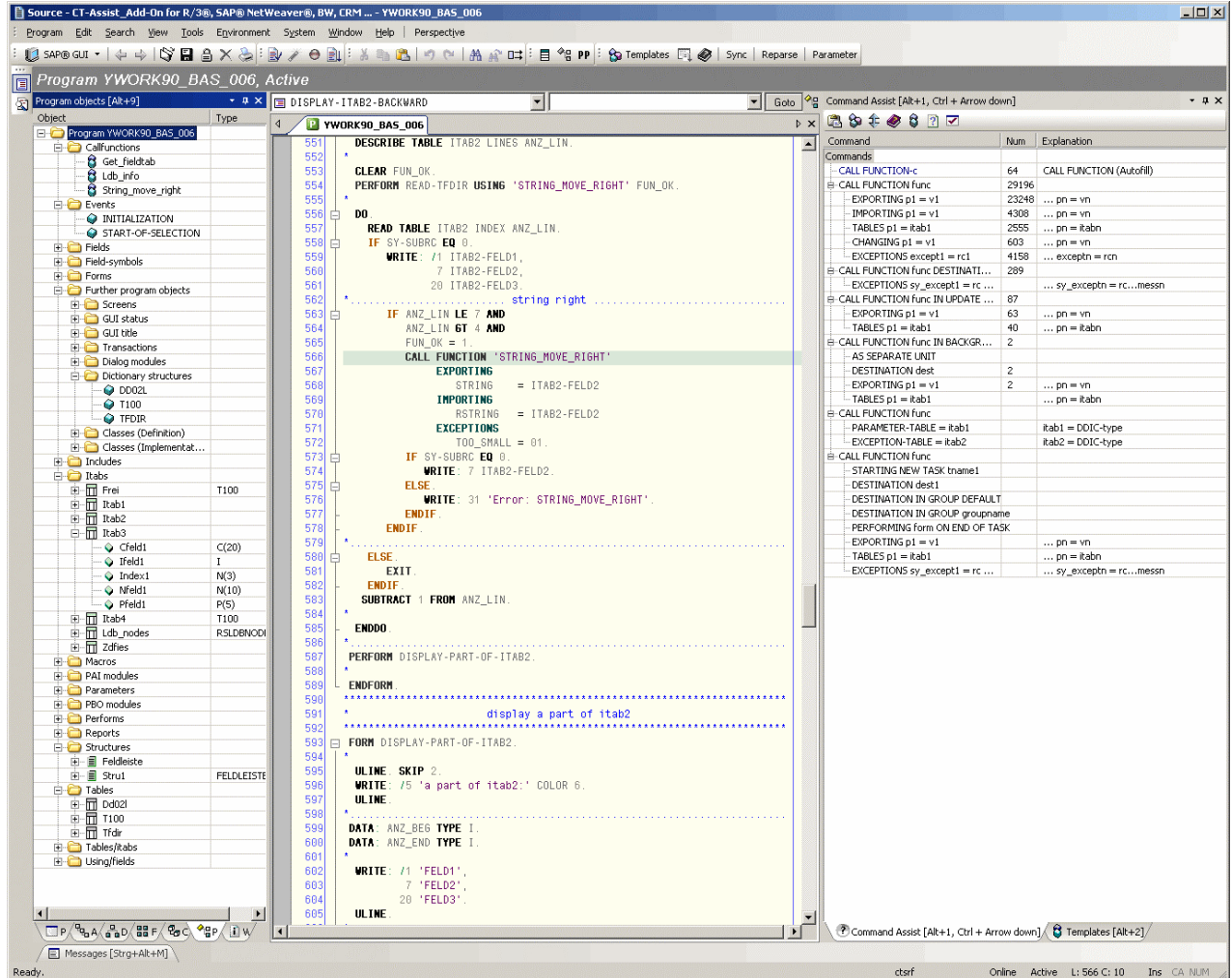
Below we will summarize the aforementioned facts to formulate a relationship between costs and benefits. Our estimates are based on a survey of 50 independent ABAP™ developers who use the CT-Assist_Module.

1.	<p>Arithmetic reduction of development costs The arithmetic reduction of development costs is 195 (120 + 60 +15) minutes per working day (see above). Does this seem high? Remember that 120 minutes were already measured by using statistic information. Only an assumption that the programmer is using the conventional editor 4 hours per day is relative.</p> <p>You might not agree with a value of 195 minutes for the cost and benefit analysis, so we will reduce the benefit to 90 minutes per working day.</p>	90 minutes/day
2.	<p>Estimate the qualitative improvement of source code We have estimated nn minutes per day. Please, decide for yourself, how much time could be saved here, keep the lifetime of your software (maintenance) in mind.</p>	-----
3.	<p>Saving per month = 30 hours 20 working days x 90 minutes = 1.800 minutes = 30 hours</p>	30 hours
4.	<p>Cost reduction per month = US\$ 1,500.-- ABAP™ development costs (internal) = US\$ 50/h x 30 hours</p>	US\$ 1,500.--
5.	<p>Benefit for the 1st month = + US\$ 500.-- Cost reduction/month = US\$ 1,500.-- minus initial costs of the CT-Assist_Module = US\$ 1,200.--</p>	US\$ 300.--
6.	<p>Benefit for 2nd - 12th month = +US\$ 16,500.-- (cost reduction/month = US\$ 1,500.-- x 11 months)</p>	US\$ 16,500.--
7.	<p>Overall benefit in the first year of purchase</p>	US\$ 16,800.--

Please give us another tool for R/3 ® which offers a potential to save approx. 90 minutes per day and at the same time helps to improve the ABAP™ program quality.

A screenshot ...

The following screenshot shows the standard GUI of the CT-Assist_Module. You can change the placement of all most all GUI elements (e.g. menu, icons, message lines, pop-up windows, keyboard layout etc.).



If you zoom the Acrobat® Reader® to 200% you can see the details.

Contact

CT-Softwareberatungs GmbH
Ziegeleiweg 8
33415 Verl
Germany

Web: www.ct-software.com
Email: sales@ct-software.com
Tel: +49-(0)5246-9310-15
Fax: +49-(0)5246-9310-16

Copyright © 2006 CT-Softwareberatung GmbH. All rights reserved. Microsoft®, WINDOWS®, NT®, XP® are registered trademarks of Microsoft corporation. SAP®, R/3®, mySAP™, ABAP™ are trademarks of SAP AG. All other products mentioned are the trademarks of the respective companies.