

Smart Debugger для SAP

Полный перевод описания инструмента Smart Debugger на русский язык.

Разработано на основе проекта GitHub.

Дата: 03.08.2025

□ Smart Debugger

ABAP-скрипт для отображения всех объектов и структур данных во время отладки SAP-программ.

Также работает как машина времени отладки — сохраняет все изменения переменных на каждом шаге и отображает их в удобном формате. Полезен при анализе глубоко вложенных структур, нескольких таблиц одновременно и при поиске изменений в прошлом.

⚙ Основан на технологии ABAP Script

1. Установите точку останова в любом ABAP-коде.
2. После срабатывания точки останова перейдите на вкладку «Script».
3. Вставьте туда скрипт (содержимое Z_SDDE.abap).
4. Скрипт можно сохранить и загружать повторно.
5. Выберите «Execute directly» и нажмите «Start Script».

□ Окно интерфейса

После запуска откроется окно, разделённое на несколько секций:

- Вверху — кнопки управления
- В центре — область просмотра исходного кода
- Справа — локальные и глобальные переменные, import/export параметры
- Внизу — стек вызовов, история исполнения и история изменений переменных

□ Отображение переменных

Скрипт определяет все локальные и глобальные переменные и показывает их в виде иерархии.

Можно включить отображение дополнительных переменных:

- SYST — переменные системы (sy-datum, sy-uname и др.)
- CLASS-DATA — статические переменные классов
- LDB — переменные LDB, скрытые по умолчанию

□ Иерархическое представление

Главная особенность — иерархические деревья для структур, вложенных структур и объектов классов.

Вы получаете панорамный (helicopter) обзор всех переменных на любом шаге.

□ Работа с таблицами

- Таблицы с заголовочными строками отображаются как имя_таблицы[] рядом с переменной.
- Двойной клик по переменной — открывает таблицу в новом окне.
- Можно открыть более 10 таблиц одновременно, если позволяет экран.

□ Работа с пустыми значениями

Нажмите кнопку «Initials», чтобы скрыть пустые и инициализированные переменные. Кнопка работает как переключатель ВКЛ/ВЫКЛ.

□ История и выполнение

Плагин сохраняет все шаги и изменения переменных. Затем можно анализировать данные:

- Вперёд (Forward) — стандартный режим
- Назад (Backward) — просмотр истории от конца к началу

□ Кнопки пошаговой отладки

Forward:

- Step into (F5)
- Step over (F6)
- Step out (F7)
- Continue (F8)

Backward:

- Step back
- To previous stop condition

□ Watchpoints

- Двойной клик по переменной — добавляет в список наблюдения
- Повторный клик — удаляет переменную
- Кнопка «Clear vars» — очищает список наблюдаемых переменных

⚙ Кнопки управления

- Visualisation ON — отображение выполнения программы вживую
- History ON — сохранение изменений переменных
- Depth — глубина анализа стека (0–9)
- Only Z — сохраняет историю только для Z-кода (можно переключить на Z & Standard)