# Интегрированные среды разработки программ

Интегрированная среда разработки, ИСР (англ. IDE, Integrated Development Environment или Integrated Debugging Environment) — система программных средств, используемая программистами для разработки программного обеспечения (ПО).

Обычно среда разработки включает в себя:

* текстовый редактор;
* компилятор и / или интерпретатор;
* средства автоматизации сборки;
* отладчик.

Иногда содержит также средства для интеграции с системами управления версиями и разнообразные инструменты для упрощения конструирования графического интерфейса пользователя. Многие современные среды разработки также включают браузер классов, инспектор объектов и диаграмму иерархии классов — для использования при объектно-ориентированной разработке ПО. Хотя и существуют ИСР, предназначенные для нескольких языков программирования — такие, как Eclipse, NetBeans, Embarcadero RAD Studio, Qt Creator или Microsoft Visual Studio, но обычно ИСР предназначается для одного определённого языка программирования - как, например, Visual Basic, PureBasic, Delphi, Dev-C++.

Частный случай ИСР, их эволюционное развитие — среды визуальной разработки, которые включают в себя возможность визуального редактирования интерфейса программы.

Интегрированные среды разработки были созданы для того, чтобы максимизировать производительность программиста благодаря тесно связанным компонентам с простыми пользовательскими интерфейсами. Это позволит разработчику делать меньше действий для переключения различных режимов, в отличие от дискретных программ разработки. Однако, так как IDE является сложным программным комплексом, то лишь после долгого процесса обучения среда разработки сможет качественно ускорить процесс разработки ПО.

Обычно IDE ориентирована на определенный язык программирования, предоставляя набор функций, который наиболее близко соответствует парадигмам этого языка программирования. Однако, есть некоторые IDE с поддержкой нескольких языков, такие как Eclipse, ActiveState Komodo, последние версии NetBeans, Microsoft Visual Studio, WinDev и Xcode.

IDE обычно представляет из себя единственную программу, в которой проводилась вся разработка. Она обычно содержит много функций для создания, изменения, компилирования, развертывания и отладки программного обеспечения. Цель среды разработки заключается в том, чтобы абстрагировать конфигурацию, необходимую, чтобы объединить утилиты командной строки в одном модуле, который позволит уменьшить время, чтобы изучить язык, и повысить производительность разработчика. Также считается, что трудная интеграция задач разработки может далее повысить производительность. Например, IDE позволяет проанализировать код и тем самым обеспечить мгновенную обратную связь и уведомить о синтаксических ошибках. В то время как большинство современных IDE являются графическими, они использовались еще до того, как появились системы управления окнами (которые реализованы в Microsoft Windows или X11 для \*nix-систем). Они были основаны на тексте, используя функциональные клавиши или горячие клавиши, чтобы выполнить различные задачи (например, Turbo Pascal). Использование IDE для разработки программного обеспечения является прямой противоположностью способа, в котором используются несвязанные инструменты, такие как vi (текстовый редактор), GCC (компилятор), и т.п.

Интегрированные среды разработки также часто поддерживают пометки в комментариях в исходном тексте программ, отмечающие места, требующие дальнейшего внимания или предполагающие внесение изменений, такие как TODO. В дальнейшем эти пометки могут выделяться редакторами (напр. vim, emacs, встроенный редактор Visual Studio) или использоваться для организации совместной работы с построением тегов и задач (например, в IntelliJ). Использование комментариев с TODO так же является стандартом оформления кода на Object Pascal, Delphi. Microsoft в руководстве по Visual Studio рекомендует использовать тег TODO (наравне с HACK, UNDONE) для следующих пометок:

* добавление новых функций;
* известных проблем, которые нужно устранить;
* предполагаемых к реализации классов;
* мест размещения кода обработчиков ошибок;
* напоминаний о необходимости переработки участка кода.

Обычно интегрированная среда разработки - это совокупность программных средств, поддерживающая все этапы разработки программного обеспечения от написания исходного текста программы до ее компиляции и отладки, и обеспечивающая простое и быстрое взаимодействие с другими инструментальными средствами (программным отладчиком-симулятором, внутрисхемным эмулятором, эмулятором ПЗУ и программатором).

Строго говоря, интегрированные среды разработки не относятся к числу средств отладки. Отладка – лишь одно из свойств интегрированных сред, которые представляют собой основу любой визуальной среды разработки или [RAD-среды](https://bourabai.ru/C-Builder/RAD.htm).

При традиционном подходе, начальный этап написания программы строится следующим образом:

1. Исходный текст набирается при помощи какого-либо текстового редактора.
2. По завершении набора, работа с текстовым редактором прекращается и запускается кросс компилятор.
3. Как правило, вновь написанная программа содержит синтаксические ошибки, и компилятор сообщает о них на консоль оператора.
4. Вновь запускается текстовый редактор, и оператор должен найти и устранить выявленные ошибки, при этом сообщения о характере ошибок выведенные компилятором уже не видны, так как экран занят текстовым редактором.

И этот цикл может повторяться не один раз. Если программа имеет большой объем, собирается из различных частей, и подвергается длительному редактированию или модернизации, то даже этот начальный этап может потребовать много сил и времени. После этого наступает этап отладки программы и к редактору с компилятором добавляется эмулятор или симулятор, за работой которого хотелось бы следить прямо по тексту программы в текстовом редакторе.

Интегрированные среды (оболочки) разработки (Integrated Development Environment, IDE) позволяют избежать большого объема однообразных действий и тем самым существенно повысить эффективность процесса разработки и отладки позволяют, то есть они являются [RAD-средами](https://bourabai.ru/C-Builder/RAD.htm) различной степени автоматизации процесса программирования.

Работа в интегрированной среде дает программисту:

* Возможность использования встроенного многофайлового текстового редактора, специально ориентированного на работу с исходными текстами программ;
* Иметь автоматическую диагностику выявленных при компиляции ошибок, когда исходный текст программы, доступный редактированию, выводится одновременно с диагностикой в многооконном режиме;
* Возможность параллельной работы над несколькими проектами. Менеджер проектов позволяет использовать любой проект в качестве шаблона для вновь создаваемого проекта;
* Минимум перекомпиляции. Ей подвергаются только редактировавшиеся модули;
* Возможность загрузки отлаживаемой программы в имеющиеся средства отладки, и возможность работы с ними без выхода из оболочки;
* Возможность подключения к оболочке практически любых программных средств.

В последнее время, функции интегрированных сред разработки становятся стандартной принадлежностью программных интерфейсов эмуляторов и отладчиков-симуляторов.

Подобные функциональные возможности, в сочетании с дружественным интерфейсом, в состоянии существенно увеличить скорость разработки программ, особенно для микроконтроллеров и процессоров цифровой обработки сигналов, являющихся очень трудоемкими и труднообозримыми процессами.

## ПЕРЕЧЕНЬ ИНТЕГРИРОВАННЫХ СРЕД

|  |  |
| --- | --- |
| ActiveState Komodo  Anjuta  Aptana Studio  AVR Studio  Axure RP  BlueJ  Borland C++  [C++ Builder](https://bourabai.ru/C-Builder/index.htm)  Cloud9  Code Composer Studio  Code::Blocks  CodeLite  Codelobster  CodeWarrior  [Delphi (среда разработки)](https://bourabai.ru/einf/Delphi/index.htm)  Delphi for PHP  Delphi Sources  Digital Visual Fortran  DJGPP  EASy68K  Eclipse (среда разработки)  Embarcadero RAD Studio  Eric  [Flash Develop](https://bourabai.ru/actionscript/index.html)  [FreeBASIC](https://bourabai.ru/einf/freebasic/index.html)  Fresh (IDE)  Game Maker  Geany  GNU toolchain  HaiQ  HiAsm  IBM Lotus Domino Designer  IBM VisualAge  IntelliJ IDEA  ISaGRAF  JBuilder  JDeveloper  JetBrains MPS  KDevelop  Kuzya  Kylix  Lazarus | Microsoft QuickC  Microsoft Robotics Developer Studio  Microsoft Small Basic  Microsoft Visual Studio  Microsoft Visual Studio Express  MIDletPascal  MonoDevelop  MSE (среда визуального программирования)  [NetBeans](https://bourabai.ru/dbt/netbeans.htm)  Open Cobalt  Open Watcom  Oracle Solaris Studio  PascalABC.NET  Pelles C  PHP Development Tools  PHP Expert Editor  PocketStudio  Portable.NET  PowerBASIC  PowerBuilder  PureBasic  PyScripter  QDevelop  Qt Creator  Quanta Plus  QuickPascal  Rational Software Architect  RPG Maker  SharpDevelop  Stardust (движок)  THINK C  Turbo C  Turbo Pascal  Ultimate++  Visible Pascal  Visual Prolog  Visual Studio LightSwitch  WebLogic  WinAsm Studio  WxDev-C++  Xcode  Zend Studio |