

## GENERAL QUESTIONS

### What is Borland® Together® 2008?

Borland Together is a visual modeling platform that enables software teams to consistently deliver on-time, high value applications that meet business needs while improving quality, cost and team communication.

If you are defining or enhancing business processes, creating new applications, architecting databases, extracting design information from existing code or defining and implementing Domain Specific Modeling Languages, Together technologies keep you in sync with a common, graphical representation of requirements, architectures, and designs through visual models. Together extends team effectiveness by supporting multiple languages (Java®, C++ and CORBA®/IDL) and platforms (Eclipse™ 3.4 on all supported Eclipse platforms). By implementing and leveraging industry standards, such as OMG™, BPMN™, UML®, OCL, QVT, GMF and others, Together can support:

- **Business analysts who are defining business processes through BPMN models.**
- **Data modelers in designing logical and physical data models and exporting them into databases, as well as visualizing existing databases into ER models.**
- **IT or application analysts with the definition and analysis of requirements through UML use case analysis – the process of visualizing requirements through use cases, activity diagrams, and sequence diagrams.**
- **Architects and Toolsmiths with the definition, implementation, customization and deployment of Domain Specific Modeling Languages, consisting of MetaModels, Graphical Editors, Transformations, Reports and additional functions to support the need to express models with tailored tooling.**
- **Architects in the definition and transformation of architectures (MDA®) from UML-based platform-independent models to platform-specific models through the use of OMG standard QVT (query/view/transformation) model-to-model transformations.**
- **Development teams implementing model-driven development whereby the application code is comprised of models versus language-centric handwritten source code.**
- **Development team communication through common language and visual representation of the requirements, architecture, and code.**
- **Architects and developers to improve and monitor application quality through audits and metrics for both models and code.**
- **Analysts, architects, and developers who need to manage traceability from design and code to requirements.**

### What specific activities are typically supported by Together?

Organizations would leverage Together technologies to:

- **Improve requirements elaboration definition processes through the use of visualization and supported modeling notations such as Business Process Modeling Notation™ (BPMN) and Unified Modeling Language™ (UML).**
- **Ensure applications and/or integrations are designed and developed clearly with maximum adaptability and reusability.**
- **Ensure application requirements, architectures, designs, and code can be well understood and communicated effectively to enterprise teams without ambiguity.**

### What is the latest release of Together?

The latest release is Borland Together 2008 SP1 Worldwide which was released in April 2009.

### What Borland tools does Together integrate with?

Together integrates with Borland CaliberRM™, Borland Caliber® DefineIT™, and Borland StarTeam®.

**What version of Eclipse does Together support?**

Together was one of the first modeling platforms based on Eclipse 3.4.

**What are the key features of Together?**

Together is one of the first modeling platforms which supports domain specific languages, business process modeling, data modeling, and UML modeling for multiple platforms and multiple languages within one tool. It leverages the current Eclipse Modeling project work, for which Borland is the Co-Lead.

**What is the future of Together?**

Together will remain true to its roots: providing usable and useful modeling solutions for the latest modeling trends and technologies used by our customers. Of particular emphasis is our work and direction around Domain Specific Languages or DSLs. Borland is making significant contributions to the Eclipse modeling project(s) and will harvest these efforts in future releases.

**How do I buy Together?**

Please visit the Borland WhereToBuy page for details on purchasing Together.

**What licenses are available for Together?**

Together is offered in both named and concurrent user license models.

**Where can I find more information on Together?**

More information about Together is available on the Borland Website on the Together product page, including whitepapers, technical information sheets, demos and articles.

**Is it true that this latest release of Borland Together is no longer role-based?**

That is correct. Borland Together 2007 and beyond does not have individual Designer, Developer, or Architect editions. All features are now accessible from this single edition.

**What is the new DSL Toolkit?**

The DSL Toolkit is all about ultimate customizability. Where UML is too complex or restrictive and free flow modelling doesn't provide enough context and guidelines, it enables Architects and Toolsmiths to create EMF based MetaModels with a graphical, EMF specific UML Class diagram editor, generate Diagram Editors and complete tooling for those MetaModels, including transformations, BIRT reports and code generators and deploy a complete customized modeling solution.

**Can Designers or Developers work efficiently with this new single edition?**

Absolutely. The new product does not in any way hamper the effectiveness of the types of users Together has supported in the past role-based editions. Relative to most of our competitors, our install size is still much smaller and maintains better performance for typical tasks. Further, with this release we have the ability to turn off various feature sets so that they will not even appear in menus or run-time.

**How does Together work with Design Patterns?**

Together provides features to enable software teams to leverage best practices encapsulated as design patterns. Design patterns in Together can exist as collections of any design elements, not just classes (as is often the case). Thus, you can create your own design patterns for common use case modeling structures or other element types. In all versions of Together, you can simply select model elements and save them as a pattern for later reuse.

**What is LiveSource®?**

Simultaneous, round-trip engineering is synonymous with LiveSource. This is one of Together's hallmark features which provides visualization of source code, effortless synchronization of the model and the source code and enriches the development experience. In Together's LiveSource projects, "the model is the code and the code is the model." Since the model is built from the source code, there are no extra steps to visualize the code. And, of course, any updates to the model are instantaneously serialized to the source code.

**Can Together automatically create Sequence Diagrams?**

Yes. It is as simple as selecting a method or operation in a source code project and selecting the create sequence diagram option. You will be prompted with selections for scoping criteria based on the location of the class in the project/package structure. Scoping allows you to quickly focus on the depth and breadth of the messages and lets you manage the size and content of the generated diagram.

**What is OCL?**

Object Constraint Language (or OCL) is a part of the UML specification. This notational language lets modelers add increased precision and expression to their models through constraints and queries. Using a language to express business rules is very significant for model-driven approaches: the constraints (rules) can be expressed early on in design models and later actually used to generate code.

**What are audits and metrics?**

Audits and metrics are the backbone of the Together quality assurance feature set. Audits are analogous to automated design and code inspections. Each audit performs a single inspection such as locating unused source code, logical flaws and coding conventions. Together includes a wide variety of audits for both the model as well as the source code. Metrics automate measurement of traditional and object-oriented concepts such as lines of code, levels of cohesion, coupling and encapsulation. Using Together to quickly and accurately report these values helps to prevent errors and maintain quality throughout the lifecycle.

**What version of Eclipse does the current Borland Together support?**

Eclipse 3.4, as a matter of fact, Borland has kept its modeling products tightly aligned with the Eclipse stream – we supported 3.3, 3.2, 3.1 and 3.0 in the past. We believe that if we are going to support products for eclipse, then we need to stay current, otherwise we are asking our customers to ignore improvements made by the community.

**Does Borland Together for Eclipse work with other Eclipse 3.4 plugins?**

Yes. This is the spirit and intent of the Eclipse community. You can use the Borland hosted Eclipse Update Site to install the Together 2008 product into your already installed Eclipse 3.4, or use the Borland installer to create a local Eclipse update site (in case your team cannot access update sites outside your company's firewall). Of course you can also install the full Together 2008 product with all needed Eclipse dependencies first and then add more plugins as needed.

**What LiveSource support does Together 2008 offer?**

Java LiveSource as well as C++ and CORBA IDL modeling project types are included in this release. Java modeling supports Java5 and provides a seamless model-driven development experience. This means that you can make updates to UML 2.0 Class and Sequence diagrams and have those changes reflected in the code. LiveSource also lets you instantly visualize any other changes made to the code. For those of you who were familiar with Together Control Center (TCC) support for C++, you will be pleased to know that we have made significant improvements in the accuracy of parsing complex code. For all language projects (Java, C++ and CORBA IDL), Together includes language-specific audits and metrics to let you inspect and measure the quality of the code. Design patterns and documentation support are also included.

**What is BPMN?**

Business Process Modeling Notation (or BPMN) is a new OMG standard for business process models. The intention of BPMN is to provide a standard means for expressing a wide variety of business process models and even includes a mapping to BPEL (Business Process Execution Language). BPMN is capable of expressing simple and complex business processes using notational elements familiar to most business process modelers.

**What BPMN support does Together 2008 offer?**

Business Process project types are included in this release just as they have been in the past two releases. Creating a project of this type allows you to create a business process model using the BPMN notation in a manner very similar to UML diagramming. Element Types include:

- **Activities, Events, Message Flows, Associations ...and more.**

**What is QVT?**

QVT is the OMG (Object Management Group) standard for model-to-model transformations. QVT stands for Query / View / Transformation. This technology provides a standard language to transform UML, BPMN, data models or custom model types. Typical usage scenarios include automating transformation of a high-level design model into a more detailed model, transforming a UML model into a data model, or even transforming a Caliber DefineIT requirements elicitation model into a UML 2.0 use case model. The core benefits of this feature set are a standards-based language to express common model transformations with traceability which provide repeatable results.

**What can I do with QVT and Together 2008 ?**

QVT was created to enable transformations between models. That is, you can take an input model and transform it into another model which is usually more detailed and ready for a specific purpose. One example is to take a UML model and transform it into a data model. In this example, you might take a class model for Entity EJBs and wish to view it as an Entity Relationship diagram. In Together, you would simply use the included UML to Data model transformation project and easily render the UML model as a data model. Together can create a trace file to save the mapping between the two models. You can open this trace file and analyze the details of the transformation activities.

Borland Together's support for QVT allows the use of any EMF (Eclipse Modeling Framework) model as input or output, includes a highly usable QVT editing and debugging environment and the ability to chain complex transformations together within an Apache Ant script. Included with Together are several examples of using QVT to serialize models to structured documents such as XSD, WSDL and even BPEL.

**What OCL support does Together 2008 offer?**

OCL 2.0 support is packaged in this release including syntax highlighting, validation, and code sense. OCL is integral to many of Together's advanced features such as model audits and metrics. We also include an OCL Expression view which can be used to interactively investigate OCL expressions. Together also provides the ability to extend OCL with Java language extensions. This enables you to expand and extend the functionality of OCL.

**How do OCL and QVT work collectively in Together?**

Actually, it is the combination of these technologies that allows Together to provide the most scalable Model-to-Model transformation solution based on industry standards. The QVT language is based on OCL constructs. And since Together lets you extend OCL with native Java code, you can leverage those extensions within QVT.

**What kind of audits and metrics are included with Together 2008?**

Together 2008 provides audits and metrics for Java, C++ and CORBA IDL language projects as well as pure model-level audits and metrics. In Together for Eclipse, a full set of language-specific audits and metrics are provided. These sets are configurable and even extensible (as explained in the next item). Audits are specific for each language while metrics return values for basic project metrics such as lines of code or number of classes as well as advanced object-oriented metrics for coupling, cohesion and encapsulation. Model audits and metrics are expressed in OCL and are applicable to pure design models such as BPMN or UML.

**Does Together 2008 allow me to make my own Audits and Metrics?**

Yes. Obviously we include many general best practices for code and models, however, you are not limited to our out of the box Audits and Metrics. There is a fully documented API for source code audits and metrics. We also include a sample project which demonstrates how to create custom audits and metrics as well as details about how to deploy them within your organization. OCL-based model audits and metrics are easily extensible as well.

**Does Together integrate with requirements management tools?**

Together includes integrations with CaliberRM and Requisite Pro. The CaliberRM integration includes trace management as well as a fully-featured Eclipse client. Requirement information can be included in generated documentation and is even accessible via the API.

**Does Together integrate with the Eclipse Graphical Modeling Framework (GMF)?**

Together integrates deeply into the Eclipse environment including many features surrounding EMF. The GMF project is lead by Borland and is used as the foundation for the new additional UML2.1 Models and for the DSL Toolkit. Because the underlying metamodel for GMF is EMF/Ecore, this models can be used as input or output in QVT based model transformations.

**For what locales is Together translated?**

The commercial modeling parts of Together are translated for French, German, and Japanese. The early access DSL Toolkit is currently only in English. The underlying open source components are localized by the *Eclipse Babel Project*.

---

Borland is the leading vendor of Open Application Lifecycle Management (ALM) solutions - open to customers' processes, tools and platforms - providing the flexibility to manage, measure and improve the software delivery process.