DevSuite

Knowledge-Centric Application Lifecycle Management

Introduction

While application development conjures up many thoughts, the fundamental core lies in the knowledge gained and used throughout the development process. From informal ideas and feature requests, to formal specifications, design documents, development plans, test cases, release notes, and user documentation, Application Lifecycle Management (ALM) is a knowledge-centric process. Historically, however, development teams have managed these vast repositories of knowledge in virtual silos, with each functional group building their own knowledge isolated from the team as a whole. In today's competitive software development environment, the silo-based models will no longer support the ever-changing demands of large distributed organizations.

To truly optimize the management and development processes, distributed teams require a more knowledge-centric ALM approach that will bridge the gaps between teams and smooth the transitions during each phase of the development lifecycle. TechExcel's DevSuite was designed with this concept in mind. Every team within a development organization should interact with, and contribute to the ALM knowledge cycle. The end result is a collaborative environment that saves significant time and resources and enables the enterprise to bring better products to market faster.

The Challenge of Modern Software Development

As global demand for software continues to grow, and competition becomes more and more intense, development teams are under continuous pressure to deliver high quality products in rapidly shrinking time horizons. In order to survive and thrive, teams must be agile, strategic, and most important, adaptive with their methods. This has lead software companies to globalize their development resources, streamline their development methods, and impose rigorous quality standards. Organizations must rely on superior design, elegant implementation, and fast time to market. Achieving this goal is difficult, however, because seamless collaboration between project management and product development has become increasingly challenging with today's globally-distributed teams.

TechExcel's DevSuite enables large development organizations to realize the benefits of an integrated multi-process approach to the Application Development Lifecycle which, in turn, helps enterprises produce better products more quickly than ever before. DevSuite brings the strategic and tactical worlds together by allowing proven, management and planning processes to co-exist seamlessly with prescribed task-driven development processes.

The Development Lifecycle and TechExcel DevSuite

TechExcel DevSuite is designed to achieve the objective of balancing effective management processes and repeatable low-level development processes. The diagram below illustrates the six phases of a typical Application Lifecycle and is divided into the management and development processes, with the primary DevSuite product specified for each phase.

Management

Management processes always involve strategic decision making; for example answering critical questions about an upcoming product development project. By their nature, these processes must be flexible and to a large extent informal. Generally, management processes fall into three categories:

- Concept: What could we do?
- Strategy: What will we do?
- Planning: How will we do it?

TechExcel DevSuite fully supports these management processes with KnowledgeWise, DevSpec, and DevPlan, each of which is easily adaptable to any methodology chosen by the management team.

Development

By contrast, Development processes focus on the efficient execution of the decisions made by the management team:

- Implementation: Architecture Design, Programming, etc.
- Validation: Functional Testing, Usability Testing, Bug Fixing, etc.
- Delivery: Release Management, Documentation, etc.

These processes are driven by tight schedules, rigid requirements, limited resources, and strict quality standards. Moreover, they are more formal and structured than Management processes. TechExcel DevSuite fully supports these Development phases with DevTrack and DevTest, each of which is completely configurable to meet any methodology chosen by the development team.

The ALM Knowledge Cycle

While the Management and Development phases described above represent business processes, there is an underlying knowledge cycle that will always remain the core of any enterprise. The ALM Knowledge Cycle begins with informal product concepts and ideas that are then refined and formalized into product feature specifications. Each feature is then

directly linked to all related and supporting knowledge, scheduled and prioritized within the overall project plan, and finally assigned to the appropriate resources for implementation. The development team then begins programming work based on the project plan, with each developer having direct access to all associated knowledge using the links created in the planning process.

The Development phase continues this knowledge cycle by creating source code, development notes, use case documents, QA test cases and testing files, release notes, user documentation, and other types of knowledge items. Unifying the entire development lifecycle with a central knowledgebase enables team members to have seamless access to all knowledge related to their work. The resulting benefit is significant because not only does overall efficiency improve, but the development team has complete confidence that the final released product exactly matches the product designed by the management team.

The diagram on the next page illustrates the ALM Knowledge Cycle within the overall Management and Development business processes. As can be seen, each process, from concept all the way through validation, contains specific knowledge that is essential to current development projects but that can also be used and leveraged for future projects. Moreover, this central repository of knowledge is critical to ensuring that historical records are easily obtained for compliance purposes.
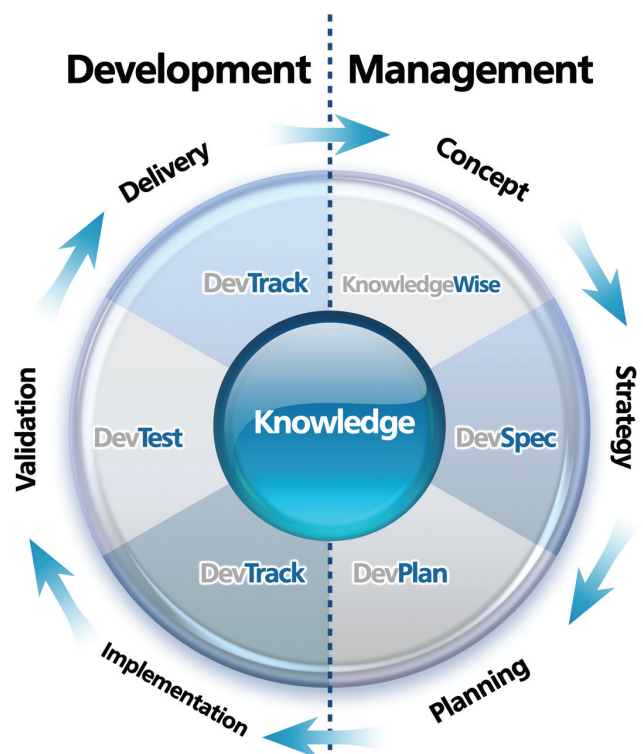
TechExcel DevSuite and the Application Lifecycle

The TechExcel DevSuite has been specifically designed to manage all phases of the Application Lifecycle. We will discuss each of these phases in more detail and review the individual DevSuite product designed to manage that phase below.

Concept (What could we do?) - KnowledgeWise

All development projects begin with ideas. Ideas could include enhancement requests from customers, feature requests from Product Marketing, design improvements from the development team, or even new product ideas from any employee within your company. This collection of knowledge/ideas is the genesis of the final delivered product.

KnowledgeWise enables the easy and efficient collection and organization of these informal ideas, gathered from a wide variety of sources using the built-in Knowledge Portal. Some of these ideas will be discarded, many will be consolidated and improved, and others will be accepted as is, but this collection, organization, and refinement of knowledge is vital to the ultimate success of any development project.

Strategy (What will we do?) - DevSpec

Good product ideas may or may not be implemented - these are strategic decision made by the product management team by balancing priorities, resources, and schedules. The strategic process of compiling conceptual knowledge (ideas) into formalized feature specifications is managed by DevSpec. More than just the refinement of ideas this formal process results in a commitment by the product management team to what features they will deliver in a final product release.

In DevSpec, a Feature is the primary element that's managed. The product management team creates a multi-layered functional tree within DevSpec, with each Feature being a leaf in the appropriate functional branch. A DevSpec Feature consists of a definable set of field-level data, including a set of system fields such as the feature's current owner, status, and due date. The Feature also includes links to all relevant KnowledgeWise items that represent the collection of ideas from which the feature was generated, as well as all other related documentation, standards, attachments, or reference items; further leveraging the knowledge-centric design process. Finally, each Feature includes a set of Events, or activity records, to formally manage the review and approval process.

The result is a Fully Designed Product which is the complete collection of features, and all related and supporting knowledge, that represents management's commitment for the final released product. Without

an effective and flexible process for creating, documenting, and communicating the Fully Designed Product, the subsequent development project will have little chance of success.

But the Fully Designed Product is never 100% final. It is the rare exception that a development project results in exactly what was defined at the start, which should be expected and, in fact, encouraged. Customer requirements change, corporate priorities change, competition changes, technology changes, resources and budgets change. This means that the strategic product definition process must use a dynamic and flexible system to accommodate, and take advantage of this constantly changing environment.

By systematically linking each DevSpec Feature to all of its related knowledge and then linking these Features to the appropriate DevPlan schedules, DevTrack issues, and DevTest test cases, organizations now have a fully integrated and dynamic system for a knowledge-centric Application Development Lifecycle. For example, if a Feature is updated in DevSpec, including modifying any linked knowledge items, each of the linked records in DevPlan, DevTrack, and DevTest will automatically be flagged with a status such as "Requires Review" and the appropriate team members notified by email. In fact, organizations can configure the system to prevent the completion of any planning, development, or testing work as long as the record remains in a "Requires Review" status. This ensures that every change to every feature specification will be reviewed by all affected groups, and that the related items will be updated.

Planning (How will we do it?): DevPlan

As the Strategy phase nears completion within DevSpec, the project planning phase begins with DevPlan. The core framework of DevPlan is the Development Sub-Project Tree. This can seamlessly share the existing functional tree structure created in DevSpec or remain independent by creating its own work breakdown subproject structure. In either case, the Development Tree leaf will always be a DevSpec Feature with all of its associated linked knowledge, allowing the planning, scheduling, and resource allocation to be managed at the feature level, not necessarily at the detailed task level.

While each feature has a set of linked knowledge defined in DevSpec, each subproject within DevPlan's Development Tree has its own set of related knowledge, globally appropriate for all features contained within that hierarchy of the tree. This knowledge could be a variety of items such as general implementation guidelines, performance requirements, industry standards, and Product Marketing documents. The combination of feature-specific knowledge, and more general subproject-related knowledge fully defines the implementation of each feature.

The Development Subproject Tree can be displayed in a standard Gantt chart view for easily defining each feature's timeline, dependencies, assigned resources, progress status, percent complete, and other project planning related attributes.

DevPlan also facilitates the collaboration and communication within the project management team throughout the entire planning process. Each Development Tree subproject is assigned a primary owner who is responsible for assigning secondary resources as needed and managing the overall subproject timeline. Another feature in DevPlan, subproject Events, allows the sub-project team to easily manage all planning-related activities such as meetings, brainstorming sessions, discussions, reviews, and approvals - easily tracked in a calendar view. This increases the visibility of critical tasks and planning milestones and allows for seamless communication between project management and the implementation team.

Implementation: DevTrack

At this phase of the development lifecycle, the project has clearly defined deliverables (DevSpec) and milestones (DevPlan). These are now handed off to the development team for implementation that is managed by DevTrack.

The ALM Knowledge Cycle

DevTrack always shares the Development Subproject Tree with DevPlan, seamlessly integrating Management and Development. While DevPlan manages the planning-related attributes of each subproject and feature, DevTrack has a completely independent interface and process definition focused solely on implementation management. The linked knowledge, however, is always shared. The clear benefit is that each developer has direct access and visibility to all knowledge related to the features that have been assigned to him for implementation, including the higher-level subproject knowledge created in DevPlan. All of this related and supporting knowledge is directly available to the developer within the feature implementation record without the need to change views or focus.

By itself, DevTrack is a very powerful, high-end, enterprise development management tool, flexible enough to handle both traditional waterfall processes as well as the newer agile methodologies. These processes can be easily modeled into any DevTrack project. In fact different teams within the same DevTrack system can even adopt different process models as needed. Regardless of the process model used, the work itself is organized and managed within the Development Tree structure established in the planning phase.

DevTrack work is managed with a completely configurable set of tracking data, team members, workflow control and automation, notification, escalation, routing, version control integration, activity tracking, QA Testing, multi-release management, and much more. To manage a single feature implementation, the development team may create many DevTrack issues, assigned to a variety of different team members, for detailed design, incremental coding work, QA Testing, and documentation. Each issue may have a set of issue-specific knowledge associated with it, including development notes, screen mach-ups, detailed design diagrams, QA test cases, and documentation files, further expanding the integrated ALM Knowledge Cycle.

In addition, because of the shared Development Subproject Tree, senior-level managers can track the current status of the project as a whole and each feature individually using DevPlan, and even view the entire list of related DevTrack issues within each subproject or feature within the Gantt Chart planning view. This data can then be used to adjust timeframes, resources, and even reallocate features to different releases as needed.

Validation: DevTest

While ad-hoc testing and unit testing is typically managed within DevTrack's development framework, functional regression testing, performance testing, and usability testing are all managed with DevTest.

As with the other DevSuite products, DevTest uses a tree structure to
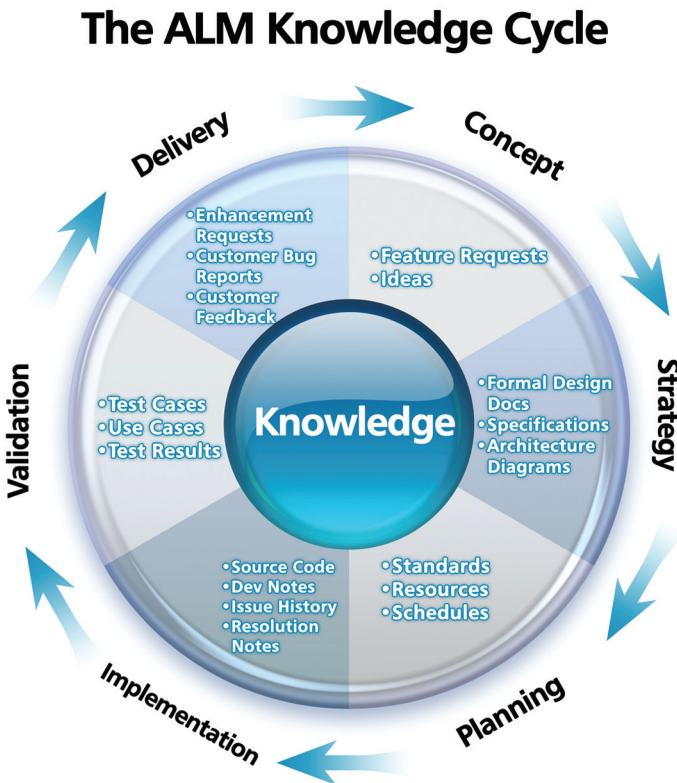
organize data, in this case Test Case Templates, which represent the master copies of detailed tests that will be executed many times for various releases and test cycles. Like the Development Tree, the Testing Tree can share the existing functional tree structure created in DevSpec, or remain independent by creating its own test case organizational structure.

Each Test Case Template is linked to a DevSpec Feature so the Test Case owner has direct visibility to all feature-related knowledge. This guarantees that the Test Case is written to validate exactly what was designed and committed to by the management team. DevTest also generates a unique set of testing related knowledge, such as sample testing documents, use cases, test automation scripts, and screen shot images, which are used by the testing team to validate testing procedures and results.

Senior Managers working in DevPlan can also view the high-level status of all QA testing (from DevTest) related to their development project within the same planning view where they track the development progress (from DevTrack) to ensure that the final release date remains on schedule. Once the programming phase of the project is complete, managers can use DevPlan to compare the final implemented version of each feature against the approved planned specification. If there are deviations, they will be reflected as DevTrack issues within that Development subproject.

Delivery: DevTrack

Products are ready for production release and delivery when the implementation and validation phases are complete. This process is easily managed by DevTrack by compiling the knowledge gained from the entire project lifecycle. Organizations can easily create release notes from the set of implemented features and bug fixes, update user documentation and even retrieve the final software product itself from the knowledge base for publishing. Finally, the collection of additional ideas and issues that were generated and documented throughout the development lifecycle is fed back into the process as KnowledgeWise items now ready for the planning of the next release to begin.

Summary

The multiple open frameworks of each product within the TechExcel DevSuite make certain that best processes and methods are used during each phase of development, while ensuring a seamless knowledge transfer throughout the process, from ideas, to strategy, planning, implementation, validation, and delivery. TechExcel DevSuite empowers today's globally distributed development

organizations to work efficiently and effectively throughout the Management and Development phases of a project, achieving a powerful knowledge-centric approach to Application Lifecycle Management.

TechExcel, Inc.
Toll Free: (800) 439-7782
http://www.techexcel.com/

3675 Mt. Diablo Blvd.	Phone: (925) 871-3900
Suite 200	Fax: (925) 871-3991
Lafayette, CA 94549	

6350 Quadrangle Drive	Phone: (919) 402-1385
Suite 120	Fax: (919) 402-1414
Chapel Hill, NC 27517	

Bei San Huan Xi Lu #48,	Phone: 011-86-10-51626938
Hai Dian District	Fax: 011-86-10-51626939
Room 19C, Building #3,	
Ke Ji Hui Zhan Zhong Xin	
Beijing, China 100086	