Understanding the applications that run your business
SETTING THE STAGE FOR UNDERSTANDING YOUR APPLICATIONS

What could be more important?

Application portfolios automate the operations of every organization. In a financial services context, for example, business processes from core banking to claims processing have become indistinguishable from the applications that execute them. As a result, it is essential that these applications align with the business strategies and goals of the organization.

A bank may launch an exotic finance product. Its loan processing applications must then be able to adapt quickly to support the new initiative. This could be by extending functionality, integrating with other systems, or perhaps by adjusting the behavior of business processes. But for some large organizations this kind of flexibility is simply not possible. They cannot adapt their applications without incurring significant cost and risk. This leads to staggering inertia and loss of competitiveness.

If this inflexibility occurred in other asset categories, it would be quickly identified and corrected. For example, the bank’s facilities management team may determine that branches cannot handle highly demanded drive-through banking. These locations would be either modernized or rationalized. Every Fortune 500 company takes a similar approach to its physical assets.

However, these same companies neglect to govern their software operations in a similarly diligent fashion. Yet, few activities could be more important. Today’s competitive environment demands agility and efficiency. It is imperative that the systems that ‘decide’ for a business on a second-by-second basis actually execute as demanded and support business priorities.

COMPLEXITY MAKES THIS DIFFICULT

Technical complexity

The reality is that existing applications have been developed over the course of years and even decades.

Documentation has become outdated, subject matter experts have moved on to other roles, and understanding of some programming languages may be unavailable. The lack of insight is made more critical as applications have grown in size to millions of lines of code and in scale as they cross environments from the mainframe to the Web. Technical complexity is compounded by a number of factors, including:

➤ Lack of modern analytical tooling to enhance understanding of applications
➤ Lack of tooling that embraces diverse application portfolios
➤ Inability to transition knowledge across teams or to new team members.

Business complexity

This decline in technical understanding is multiplied by a loss of business understanding. Business users cannot communicate what changes need to be made to reflect business goals and development teams are unable to respond effectively. This becomes especially problematic in companies with highly dynamic business models. Business complexity is compounded by a:

➤ Lack of common lexicon between business and IT
➤ Limited understanding of business functions within applications
➤ Inability to abstract business understanding out of excessive technical focus.

Disbursed intelligence

Different users have and require different information about their applications. A CIO may need to know the availability, cost, and risk of the application portfolio. She herself knows the value of parts of the application portfolio based on her strategic vision. A business analyst may need to know the adaptability of the application that runs a key business process. He also understands the structure and sequencing of business processes from an abstract level.
However, this information is rarely centralized, linked to applications, or made appropriately available to users in the development team. As a result, the absence of necessary intelligence further exacerbates the challenges of complexity. Dispersed intelligence is compounded by:

- Lack of knowledge about priorities and needs of organization
- Lack of a mechanism to connect various information sources for more complete view.

**COMPLEXITY CREATES REAL BUSINESS PAINS**

This lack of insight into existing applications leads to a variety of business problems that are manifested throughout the organization. Some of these issues will be immediately apparent as IT-related issues. Others are secondarily identified due to catastrophic business issues. The lack of business intelligence leads to an inability to:

**Transfer knowledge**

It is important that their knowledge about the structure and function of existing applications can be transferred to new team members as their peers move on or move to new roles. Similarly, as applications are outsourced, there cannot be an extended ramp-up period for the new team. Outsourcers must be productive from day one. But complexity impedes this knowledge transfer by reducing the ability of new team members to get up-to-speed on their applications.

Knowledge transfer can also be understood as a transfer of control away from individual contributors back to the management level. If managers don’t understand the inventory of their application portfolio, how elements are architected to support the business, and where opportunities for improvement exist then they cannot hope to make their application portfolios a competitive tool. Complexity prevents management from asserting control by preventing an understanding of ‘where we stand’.

**Govern the application portfolio**

Enterprise applications are critical business assets for companies that depend on complex business processes. As with any asset, it is imperative that an organization can direct effort to the most productive and important. For instance, applications that duplicate functionality, support eliminated lines of business, or automate commodity activities may be targets for rationalization.

Complexity prevents this governance by limiting insights into which applications or application elements should be targets for rationalization or modernization. It also prevents meaningful linkages of information from multiple sources – business intelligence that is vital for making informed choices about where to focus scarce resources.

**MODERNIZE TO BOOST AGILITY AND EFFICIENCY**

Frequently changed applications can become brittle and inflexible. As a result, it is risky, costly, and time consuming to adapt them to support new business processes. This inability to adapt automated business processes is a primary reason why larger and older companies are hindered from responding more nimbly to market opportunities.

Complexity is at the heart of this challenge. Management cannot determine which portions of the application portfolio to modernize or how to roadmap the modernization process.

**Adapt to business needs**

On a daily basis applications must adapt to respond to bugs, inefficiencies, or minor shifts in business tactics. This application maintenance requires coordination between the business users or managers that request changes and the – often remote – development teams. Business users must be able to formulate change requests that can be communicated.

Developers must understand the code well enough to know what changes must be made to reflect the request without creating unintended impacts.

Complexity in a maintenance operation leads to costly breakages, poor performance, and rework. Both sides of the maintenance workflow must have accurate, current, and shared insights into their application portfolio in order to facilitate the change request process.
What business intelligence should be collected?

But what is the business intelligence that must be made available to an organization in order to confront complexity? The categories closely match the related sources of complexity.

Technical insights

Development professionals must be able to understand the complex structures and behaviors of their enterprise applications. This requires visualizations of how data and other elements flow through a system. It requires the ability to probe an application for artifacts of interest, such as variables that require extra security, elements that will be affected by a proposed change or code that is inefficiently architected. It requires the ability to measure the size and complexity of the applications to assess the size of a project and focus effort.

To fully understand the reality of our application portfolio requires rich technical insight. Rather than relying solely on high-level scans of artifacts, users must be able to assess their applications to a level that is useful. While a manager or architect may find summary analytics to be sufficient, a developer will require highly detailed analysis within programs in order to execute needed changes.

Similarly, users will require technical insight across diverse applications. Application portfolios consist of software from COBOL to Java. Technical analysis must be just as inclusive.

Business data

Over time applications often lose their connection with the business function they were designed to support. This is manifested in the creation of names for variables, programs, and other elements that are impenetrable to business users. This requires a shared lexicon that allows business analysts and development teams to speak the same language.

Additionally, it is important for users to understand the logic of how a system behaves. This logic describes business decisions like “approve discount if customer status equals ‘gold’”. Governance of this logic allows analysts to better communicate how a process must adapt to support new goals. It also allows development teams to focus immediately on adapting the correct logic.

Stakeholder insight

As discussed, users have different information and information needs related to the application portfolio. Complexity can be reduced by collecting relevant insights and making them available where necessary. For example, a business user may have insight into the purpose and behavior of a business process. Mapping this intelligence onto software assets allows development staff to more readily understand how code modifications should be executed.

Other kinds of information can also be associated with applications to provide a richer insight. For instance, executives can assign value and risk to portions of the application portfolio. Similarly, cost and business service data like availability could be associated with an application. This metadata requires a mechanism for associating it with the underlying applications.
Stakeholders have different levels within and different functions across an application. This means that their information requirements will be different. A CIO will need a portfolio-wide view of her applications. A developer will need a deep view of a particular portion of an application. Similarly, a manager of a business process will require information about, say, availability and complexity, to be process-centered. The manager of an outsourcing relationship will require information about adherence to service level agreements to be provider-centered.

This mechanism for grouping intelligence and abstracting it to the appropriate level is business contextualization.

**BUSINESS CONTEXTUALIZATION FACILITATES BUSINESS INTELLIGENCE FOR APPLICATIONS**

Business contextualization is the concept of overlaying meaningful groupings onto applications. That is, it connects ways of describing and organizing the business to applications and their constituent artifacts. This reduces complexities and allows users to view their portfolio from meaningful business perspectives. Contextualization is achieved by allowing users to ‘tag’ or group their software artifacts based on concepts that are useful to the business. For instance, users may wish to understand how their software is structured to support business processes. We could tag artifacts based on how they relate to our business process model. Another user may need to understand which geographies control a given set of software. Similarly, we can overlay geographic groupings based on where software is managed. Users can quickly construct hierarchical tagging architectures and overlay them on their software assets. This context can be used in many ways. Users can:

**Understand applications as business assets**

Technical diagrams are useful for executing technical activities. But when managers seek to understand how their operations are instantiated in an application, they must be able to view them from a higher level. Business contexts enable users to ‘abstract’ their views to see how software is grouped into concepts that are useful for the business. Managers can instantly see relationships and dependencies between functional areas, development teams, and the like. This allows them to visualize how a system could be better architected to best support business needs.

**Focus effort on assets that matter**

Business contexts allow managers to focus resources just on portions of their application portfolio that matter. Managers can constrain their analytics to targeted contexts and development teams can drill down within context hierarchies to further focus just on software that relates to a given initiative.

Users can answer questions like: where are architectural inefficiencies in my important business processes? Which code should adhere to Sarbanes-Oxley requirements? Which code should comply with stricter security requirements? Which code is managed by a certain service provider? Where should I replace portions of my portfolio with packaged or redeveloped applications?
Manage applications as business assets

CIOs should always make portfolio decisions in the light of business priorities. Technical metrics like ‘lines of code’ and ‘cyclomatic complexity’ of an application are often poor guides on their own for IT decisions. They must be combined with other measures and associated with the key performance indicators (KPIs) that the business uses for decisions. Technical measures take on resonance when demonstrating the inflexibility of a business process or the poor performance of a service provider. They are still more powerful when grouped with other measures like cost, risk, and value. Business context provides this mechanism.

HOW IS THIS BUSINESS INTELLIGENCE COLLECTED?

Scope the application portfolio

Many portions of the application portfolio are static or not strategic. Expending resources to understand these areas is not a wise investment. We should begin by focusing on highly dynamic and mission-critical applications. We can then focus our intelligence gathering on these areas.

Contextualize the target area

Much of the contextualization process can be automated. Regardless, it is still important that more thorough contextualization is concentrated on areas that matter. The context maps that are overlayed onto the application portfolio could be defined in business process or enterprise architecture modeling tools and then reused for context maps. We should create context hierarchies upwards and downwards to the levels at which we want to manage our portfolios.

Assess the applications

We load the targeted source code into our application knowledgebase and collect detailed technical information about the application. Information like metrics, visualizations, and the data necessary to respond to queries is collected.

This information can be enriched by metadata from subject matter experts, for instance information about the business logic embedded within the targeted contexts.

Launch the business intelligence repository

Technical and business knowledge has been collected, and can be refreshed by reparsing code and continuing to import data from stakeholders and third-party software. Now, it is time that the information is consumed in order to enhance alignment.

HOW IS THIS INFORMATION CONSUMED?

This information only becomes true business intelligence when it is consumed. These actors make use of insights into the application portfolio to perform one of the solutions described above. Each will consume information in a different fashion for different goals. Let’s turn to the three ways in which the intelligence is utilized.

Assessment and management reports

Summary inventories and assessments provide managers with a foundation for planning modernization and development activities. For instance, management may identify ideal candidates for application renovation in order to improve architectural quality.

Similarly, portfolio management dashboards allow managers to track KPIs over time. This allows executives to determine where misalignments exist between goals like availability or efficiency and the reality of their applications. Complexity is mitigated by identifying priority development tasks and redirecting resources away from uncritical activities.

Development teams

As discussed previously, business analysts and development teams are heavy consumers of the collected intelligence. They use the always-current, always-shared information to better communicate business requirements and technical challenges. The information is also vital for executing daily development tasks and more sophisticated modernization initiatives. Developers can understand the targeted applications, locate potential impacts, create test cases, and more. This is especially compelling when matched with ITIL principles for improving application management.
PLATFORM ALLOWS INFORMATION TO BE CONSUMED INTELLIGENTLY

To address these needs, the world’s most demanding development organizations have relied on Micro Focus’ Modernization Workbench®. This provides users with centralized business and technical insight into their core applications. This allows global IT teams to identify, analyze, and control development activities that realign applications with business requirements.

The Modernization Workbench is designed to be extended. Different information sources can populate information into its knowledgebase. This information can be filtered and adapted to suit the needs and methodologies of information consumers, like business or development managers. Let’s first look at the ‘intelligence sources’ side of the equation.

Information can be added to the repository from a broad number of sources. But which information is added and how it is added will differ by implementation site. The platform analyzes existing applications – from COBOL to Java – and extracts detailed technical insights into the structure and complexity of the application portfolio. Information from SMEs, or models from other sources, like industry frameworks, can be inserted into the model to serve as a container or context. Further, information from third-party tools such as Business Service Management tools or Application Lifecycle Management tools can enrich the information about an application that has been stored. Additionally, questionnaires to survey opinions of subject matter experts can be adapted to suit the methodologies of system integrators and used to collect data from SMEs.

Once information is stored in the knowledgebase, it can be consumed in different ways by different consumers. Users can readily build content onto the platform to adapt how the information is outputted. For instance, a user may have a unique formula for combining KPIs to yield an assessment report or APM dashboard.

CONCLUSIONS

The business and technical complexity of applications often means that organizations cannot adapt quickly enough to respond to new strategic goals. To confront this challenge, the best practice is to collect targeted technical, business, and stakeholder information and present it in meaningful contexts to the right users in the organization. Armed with this information, business users can allocate IT resources toward business priorities, and development teams can execute realignment activities more proactively.

About Micro Focus

Micro Focus, a member of the FTSE 250, provides innovative software that allows companies to dramatically improve the business value of their enterprise applications. Micro Focus Enterprise Application Modernization, Testing and Management software enables customers’ business applications to respond rapidly to market changes and embrace modern architectures with reduced cost and risk.

For additional information please visit: www.microfocus.com

© 2011 Micro Focus IP Development Limited. All rights reserved. MICRO FOCUS, the Micro Focus logo, among others, are trademarks or registered trademarks of Micro Focus IP Development Limited or its subsidiaries or affiliated companies in the United Kingdom, United States and other countries. All other marks are the property of their respective owners. sWPUTAT0811