

A Roadmap to Productivity



Application Relationship Management

by Peter Szirmak
Senior Partner

Executive Summary

This paper describes Application Relationship Management, a new paradigm in Information Technology. To quickly position the A.R.M. concept, a comparison will be made to Customer Relationship Management (CRM), a well-known concept by today.

CRM organizes activities of an enterprise to implement a customer focused business strategy. Today, a variety of software solutions exist that implements such a business strategy, by providing connectivity between the various business functions. In essence, CRM solutions facilitate the understanding of relationships within and across various business functions, all relating to the customer.

While organizations quickly came to accept the importance of such higher level understanding of business operations (with the customer being the common thread), nothing exists that would enable IT organizations to have a **connected view** of their IT assets.

At the same time, today's IT environments have become increasingly complex to understand and oversee. Company mergers and acquisitions have further added to the complexity of IT environments, already containing a myriad of (both internal and vendor supplied) software, hardware, documentation, best practices and other types of information sources.

At best, companies have a level of understanding of their physical environment (e.g. a given application runs on a given hardware) or some very high level documentation of how business systems relate to software assets (e.g. a given business system contains certain applications).

Organizations would also typically have good detail understanding of their various IT assets (for example, they may have implemented a document management software or specific application mining tools for specific platforms) but each area is separately managed by the enabling software. This state is very similar to where organizations were before the concept of CRM was introduced by the Gartner Group: the various business functions utilized tools for their own areas (such as contact management software, accounting packages) but there was no overall connectivity between these tools.

Application Relationship Management (A.R.M.) implements a knowledge-centric IT strategy that enables the productive management of IT assets, by providing an understanding of relationships within and between all IT assets. It provides a dynamically and continuously maintained, connected view of IT assets. In a sense, A.R.M. is the 'CRM for the IT world'.

A.R.M. allows the organization to continuously understand its assets: what assets exist, what assets are used/unused, how those assets relate to each other, and what relationships exist within the assets.

Application Relationship Management defines a strategy to create and use an overall roadmap for an organization's IT assets, including documentation, application source code, business objects, hardware and all other components. The power of such a roadmap is clear. It provides organizations with a strategic advantage over competitors.

The importance of implementing Application Relationship Management cannot be emphasized enough. Literally all aspects of Information Technology benefit from such a strategy, from everyday development and maintenance activities to strategic platform migration, application integration, and outsourcing or repatriation projects. Literally any IT project hinges on understanding what one has and how it needs to change. The ability to effectively analyze information in a timely and quality manner, directly relates to overall IT expenses.

To ensure that business objectives can be met on a repeatable and cost-effective basis, IT organizations need a connected view of their IT assets. To measurably and permanently reduce costs associated with the planning, designing and executing of IT projects, organizations need an **automatically generated knowledge repository**

that can answer such simple questions as: 'What do I have?', 'How do my assets relate to each other?' 'How do they depend on each other?' or 'What is the impact of changing/removing/adding components?'

Relationship information is inherent but currently hidden in IT assets. Application code, hardware, documentation, etc. all contain relationship information within them. Application Relationship Management implements a strategy that allows organizations to automatically pull such information to a higher, connected level. It also defines how such information is dynamically maintained.

This paper outlines what Application Relationship Management is, its scope and purpose, as well as outlines suggested implementation scenarios.



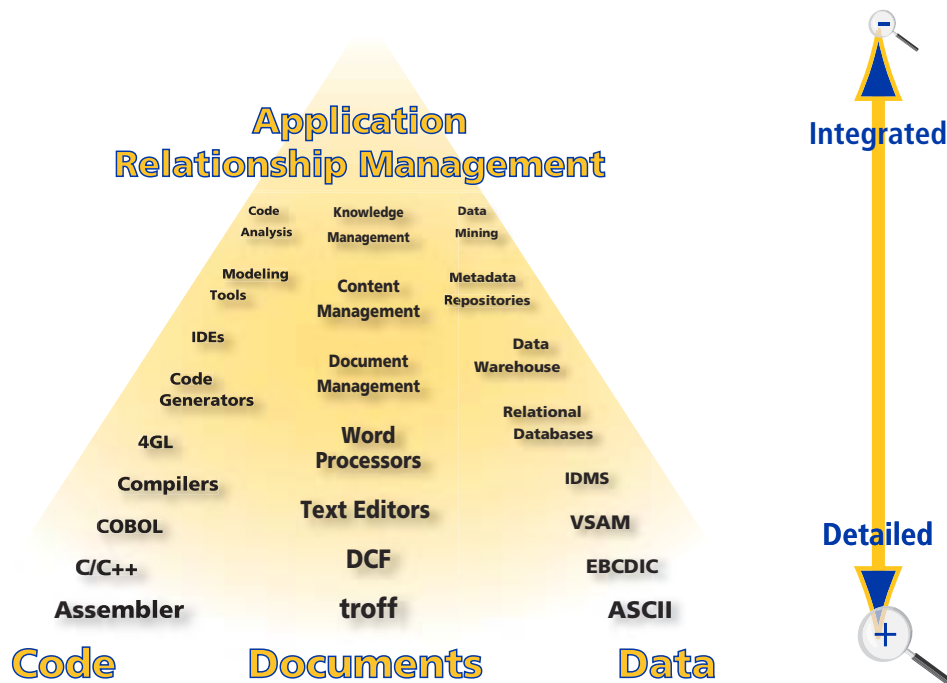
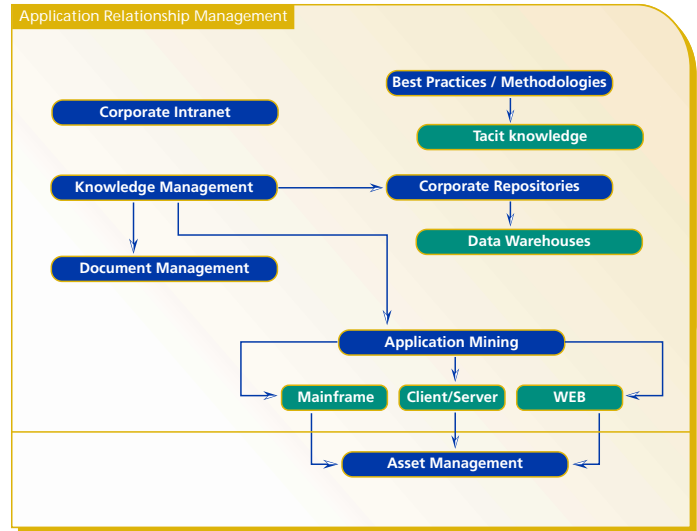
What is Application Relationship Management?

Application Relationship Management provides a roadmap for creating and maintaining a central repository of all your IT assets. It sits on top of other initiatives such as Knowledge Management or Application Mining, by connecting all these and other areas of the IT organization.

The ultimate success of individual initiatives such as Document Management and Application Mining will depend on how well these areas connect to each other. Having a solid document management process by itself will not answer the questions: 'how do my documents relate to what I have?' - or, 'which documents contain relevant information?'

Similarly, leading Application Mining tools such as CAST and Revolve will provide the organization with detailed information about their various application platforms. They will, however, not answer the question: 'how does my client server application relate to my mainframe applications or to the documents and other objects I have?'

The figure below depicts the scope of A.R.M.:



Application Relationship Management can be looked at as the evolution from detail to higher level understanding, as depicted here:

Current State

'The inability to provide the right knowledge to the right people at the right time is expected to cost Fortune 500 companies over \$57 billion over the next two years' (International Data Corporation).

This widely quoted statement summarizes well-known realities in our industry. Slow, costly execution of projects results in delays for bringing new business initiatives to market. The Standish Group Chaos study for the year 2000 found that 23% of projects outright failed and 49% were challenged. Of these challenged projects, the average project was 189% over budget, 222% behind schedule and delivered only 61% of originally specified features.

Silos of Information

For IT initiatives, the amount of information to be looked at and analyzed is growing at an exponential rate. At the same time, information is decentralized, kept in different formats at different locations.

Corporations installed software that organizes information on source code, documentation, execution statistics, business models, etc. However, handling software for each silo of information is specific to the area it handles: application mining tools will provide information about specific source code on specific platforms; modeling tools will organize process and data models, document management software target documentation only, etc. None of these tools was designed to communicate with other areas where related information exists. Such relationships are left for manual research and understanding.

The time and effort spent in information gathering is directly related to the accessibility of information. Typical analysis will need to gather information kept in people's head, paper format, on the mainframe or on the PC, in various formats. Some information is in completely raw format (e.g. source code) or kept in the tool organizing the given information (e.g. application mining tools).

Lack of Qualified Resources

The proliferation of knowledge management and other tools, all intended to organize and maintain these various silos of information, necessitate that IT professionals become 'jack of all trades', understanding multiple environments and tools. The tools themselves are involved and complicated tools, each requiring significant experience and understanding.

In almost every IT organization, initiatives exist to attract, train and retain qualified resources. However, having deep understanding and knowledge of how IT assets relate to each other are usually the privilege of a few. These resources obtained their knowledge through many years of working with these assets. The deep knowledge of these individuals is typically not captured anywhere.

As a logical result, IT organizations unduly rely on the knowledge of a few key individuals. These resources become the bottleneck in initiatives. As a result, all projects count on a slower than necessary delivery and experience costly delays.

Having knowledge centralized in the head of a few individuals also causes a slow and costly training of new resources. Building organization-specific IT knowledge takes many months if not years. New resources become truly effective much slower than desired.

Mis-Utilized Tools

Corporations spend significant amounts on purchasing and maintaining tools that provide organized information on specific areas of IT. At the same time, tool vendors struggle to keep their tools utilized by their clients but more often than not, after a relatively short period of time, many of these powerful tools become 'shelf-ware'.

The underlying reality is simple: the tools themselves are designed for specialists but are purchased for general use. The information available via these tools is extremely important for specialists but is much more than most IT professionals need on a regular basis. Lacking any other way of getting at the information, corporations purchase a much larger number of licenses than they truly

need, in the hope that occasional users will use the tool for getting the information they need. In reality, occasional users become non-users and return to old-fashioned, mostly manual research (once the effects of the original training fade, the tools are perceived as complicated and unfriendly). This results in a large number of licenses becoming 'shelf-ware'. Typically, after some time, the overall return on investment is judged to be low and all licenses are terminated, resulting in the return to manual analysis for all. Soon after, a new tool is purchased, starting the cycle again...

Desired State

The Gartner Group predicts that IT expenses (as a percent of overall corporate expenses) will grow from 30% today to 50% by the year 2010. Such expenses will continue to be challenged, resulting in progressively severe budgetary pressures for IT.

Delivering more with less requires an overall strategy for productivity increase. Current initiatives of document management, application mining, etc. need to be connected in order to produce the desired productivity increase. If an overall strategy does not exist, these initiatives may only add to the creation of information silos and other problems discussed above.

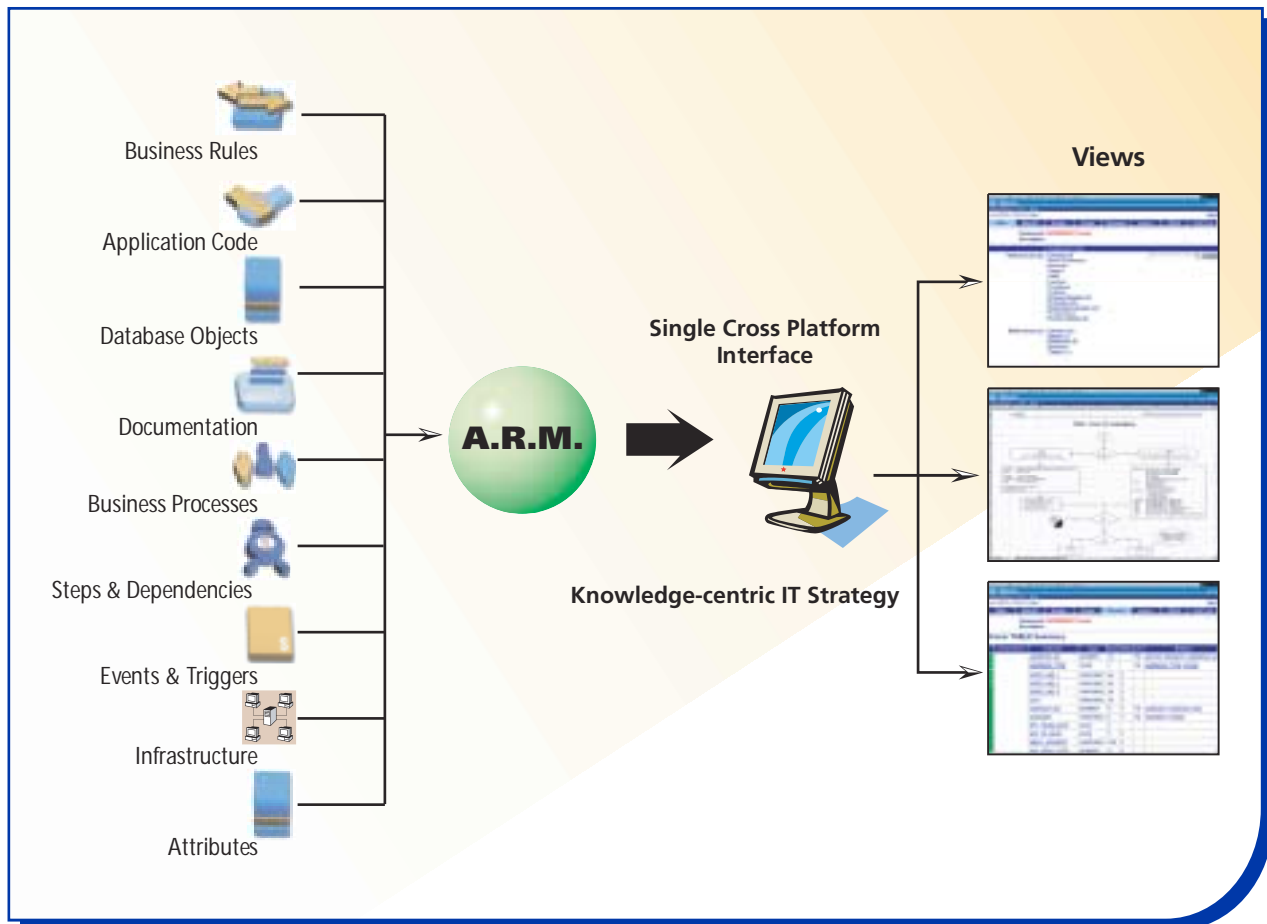
Relevant tools may continue to organize and manage specifics of a given IT asset. However, A.R.M. implements a repository of relationships between and within those assets, thus providing a much needed higher level understanding of connections and dependencies. A single interface (preferably via the corporate intranet) establishes the gateway to all such IT information.

Regardless of the actual location or platform of the information, IT professionals will have a single platform to use.

Relationships within and between IT assets will no longer be the hard-earned knowledge of a few but would be automatically

generated into the A.R.M. repository, for all to see. Business rules buried (in code or documents) become visible to all.

Existing investments into tools will provide a much greater return. Specialists may continue to use the given tool but information required by occasional users is automatically pulled a level up into the A.R.M. repository. By doing so, fewer licenses will be required but the given tool becomes an integral part of the organization's information infrastructure and thus will continue to be used. By implementing a properly utilized tool environment, both IT organizations and tool vendors will ultimately benefit.



The figure above depicts a knowledge-centric IT strategy, as defined via A.R.M..

Benefits of Application Relationship Management

Implementing Application Relationship Management will result in significant increases in operational efficiencies, resulting in conservatively estimated cost savings of 10-20% of IT budgets related to projects.

The key to Application Relationship Management is changing the current, static knowledge capture and maintenance to an automated, dynamically maintained process, thus providing the following key benefits:

- Protect IT investments by ensuring their cost-effective usage and evolution
- Standardize IT asset information, regardless of original platform or format
- Enable knowledge proliferation across the IT organization

Application Relationship Management will result in tangible cost savings in practically all areas of the IT organization, from strategic planning through development, integration and enhancement projects to everyday on-call maintenance. Three specific areas of the Systems Development Lifecycle will see significant benefits: Estimation, Analysis and QA/Testing.

With A.R.M., estimation efforts will be more accurate and many times faster than currently possible. Such efforts currently necessitate extensive manual efforts of a few. With A.R.M., information is available in actionable format to many.

Reducing dependency on a few bottle-neck resources and at the same time facilitating the effective analysis of applications, documentation, etc naturally results in cost savings at the front end of projects. Well-accessible and correct information will also enable the creation of more effective test plans and overall reduced costs in ensuring the success of QA and Testing phases. The overall quality of project output will increase.

Organizations embarking on Application Relationship Management will also see reduced training costs, both formal and informal. Utilizing the corporate intranet to access information reduces the need to train in specific, more complicated and diverse tools. At the same time, by automatically pulling information inherent in existing assets to a higher, common level, the need for informal training (new resources trained by existing personnel) will be reduced.

Implementation

Naturally, one can opt to a large scale implementation of Application Relationship Management, with a phased approach to drive maximum benefit. When added up, the benefits of complete understanding (via an automatically generated, central repository of information) of one's IT portfolio are extremely large. However, the highest likelihood of success is delivered by an implementation plan that addresses current pains, critical projects and overall business drivers alike, and takes a realistic ROI focused approach.

The IT industry has gathered valuable experiences during CRM implementations (many of them proving costly and some outright failures). Based on these experiences, the suggested approach contains four major stages:

Stage 1: **Requirements Definition & Prioritization**

In this phase, the priorities and goals of the organization are understood and documented. Based on information gathered, a customized implementation plan for Application Relationship Management is created: in essence the **Roadmap to A.R.M.**.

Deliverables of this phase include:

- Requirements Document, defining business objectives and priorities
- Scoping Document, defining the breadth and depth of the repository (for Business Processes/Functions, Best Practices, Business Systems, Infrastructure, Documentation and Data)
- Implementation Plan, defining exact timelines and costs.

Stage 2: **Focused-scale Implementation**

In this stage, a relatively **smaller area** of the IT organization is chosen to implement A.R.M.. The suggested approach is to identify a project or area of the IT organization **that immediately benefits from** the availability of **A.R.M.**. This can be the maintenance area of the organization or practically any project such as platform migration, integration or major enhancement projects where understanding the content and relationships of the IT portfolio is crucial.

The goal is to selectively implement A.R.M. for the chosen area with specific measurements in place to solidify the business case for large scale implementation, as well as to pilot the implementation process. Some components of the implementation (such as user awareness training, etc) can also be piloted for maximum results. The automated maintenance of the A.R.M. repository (with hooks into the existing company environment) can also be implemented at this time.

By ensuring immediate benefits for the selected area, the costs of the pilot can be justified. Rather than investing in 'trying out A.R.M.', one should implement A.R.M. with a specific cost-saving goal for the chosen area, thus financing the 'pilot' itself. Note that many CRM implementation failures were caused by 'trying to boil the ocean' - investing large amounts without tangible and measurable benefits along the way. The failures were not caused by the inappropriateness of CRM itself. These projects failed because the implementation plan was driven by hype rather than corporate reality.

Stage 3: **Large Scale Implementation**

In this phase, A.R.M. is **implemented across the organization**, as per the A.R.M. Roadmap. Sub-phases may define the order in which various areas implement A.R.M..

Experience gained in the previous phase allow for an effective and timely implementation.

Stage 4: **Repository Maintenance**

Finally, the **A.R.M. Repository is maintained** on an on-going basis, via automated procedures customized and finalized during implementation.

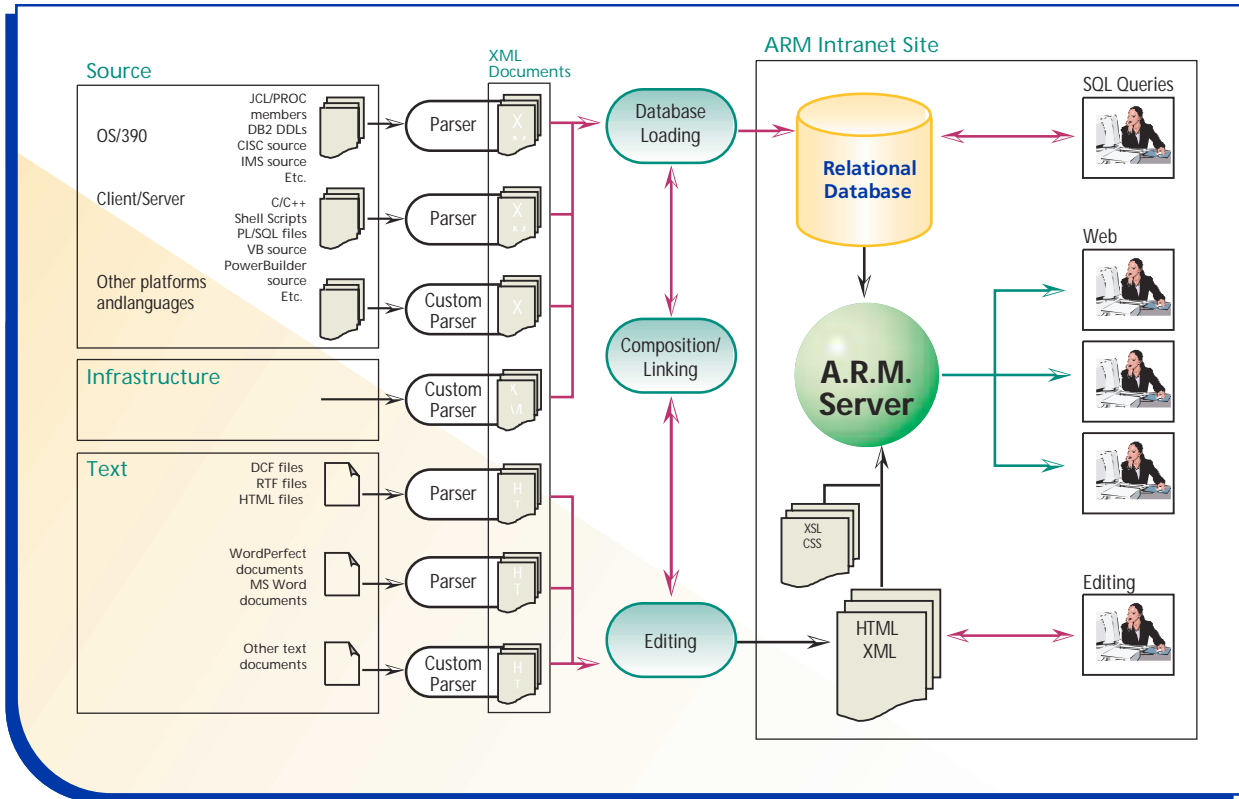
Typically, a Service Level Agreement will define the frequency and level of support maintenance required for various areas of the IT organization.

Since almost all maintenance is automated, typical costs for maintenance should be extremely low compared to the benefits realized.



A.R.M.

Over the past many years, technology has been developed at our site that effectively implement Application Relationship Management.



The diagram above shows the technology process behind A.R.M.