Phios Corporation

White Paper

The Phios "Whole Product Solution" Methodology

Norm Kashdan

Phios Chief Technology Officer

1 Introduction

The senior staff at Phios has several decades of experience delivering software solutions to industrial clients. During this time it has become clear that delivering software "applications" is only a small part of any complete solution. Often, a successful implementation needs to take into account infrastructure, deployment, support, business processes, and underlying content knowledge. Further, any solution must be based on a deep understanding of the different roles performed by persons that use the applications, and in particular, those roles required to maintain the underlying knowledge. Consequently, Phios has developed a delivery approach that addresses the "whole product solution".

A further realization is that much of the logic, content, and process flow comprising a solution already exist in the subject matter (business) domain. Therefore, in properly formulating the solution, this knowledge has to be gathered and made explicitly expressible in a language that is: completely familiar to subject matter experts and users, able to be used directly by software applications, and maintainable by non-programmer "knowledge workers". In recognition of this, Phios has evolved a knowledge-based approach to solution development that distinguishes and separates knowledge from the applications that use it. Phios solutions provide knowledge tools and applications specifically designed for maintaining this knowledge.

This white paper describes the characteristics of a knowledge-based solution, as well as the Phios "Whole Product Solution" Methodology used to produce such a solution.

2 Anatomy of a solution

In today's business world, any software solution invariably lives within a complex environment of other applications, systems, and infrastructure. Rarely is it possible to build a "standalone" application anymore. Given the nature of such distributed environments, it is also not desirable to trap business knowledge in specific applications, especially if such knowledge needs to be available more widely, and can only be maintained by software programmers.

It is critical that a modern software solution take these factors into account. Phios has created a solution approach that recognizes this reality.

2.1 The Whole Product Solution

When we refer to a "whole product solution" we refer to everything it takes to "solve the problem", "get the job done", and, "put the solution into production". This often means dealing with numerous issues beyond merely building a software application. This also means not only solving today's problem, but building in adequate flexibility for the future, since we know that all businesses and systems evolve and change over time.

Any solution starts with a consideration of the business problem to be solved. Who, and in what roles, needs to do what? Who needs to inter-operate with the software solutions? What business processes are affected? What needs to be changed in order to accommodate the new solution? What external systems supply and demand data from the new "solution"?

Then, there is infrastructure to consider. What other technologies and platforms are involved? How does the new solution operate with the conditions and constraints imposed by the solution's proposed environment?

Finally, and most importantly, there is knowledge – or, as we often refer to it, "content". What information does the solution use? How, where, and by whom does this information get assembled? How does information change over time and how does additional information get added? How is this information, or content, made actionable within the proper contexts, effectively turning it from information to knowledge?

All of these questions indicate the breadth of considerations that need to be addressed in creating an industrial software solution.

The Phios approach to solution development takes all of this into account. Our development methodology assures that any solution we create is a "whole product solution".

2.2 Application Content

Most non-trivial application solutions have a rich complement of content that is best organized and maintained by subject matter experts and users, not programmers. It is important that in any modern solution this requirement is factored into the solution. For example, a particularly interesting type of content we often deal with is branding. Branding refers to the need for an application to change its look, and sometimes its behavior, based on who uses or sees it. Branding requires the management of brand content. It is difficult, expensive, and generally inadvisable to rely on programmers to maintain brand content.

In a Phios supplied solution we segregate as much of the business content as possible from the actual application code. This business content is formed into "configurations" that are used by the application. In a properly designed application these configurations can alter the appearance and behavior of the application without requiring coding changes. Phios solutions are designed to allow such configurations to be maintained without the use of programmers.

Phios provides the ability to maintain content by subject matter experts and users in three different ways:

- Knowledge studio tools designed specifically to add and maintain content
- Role-based applications to support designated content management roles that maintain content
- Content management modes within the application solution that allow users to maintain content

The specific approach is dictated by the needs of the solution and its users.

2.3 The Double Cycle Solution

Because of the distinct separation of content and application, a Phios style solution is generally organized into two cycles of work.

2.3.1 Content Management Cycle

The least common and less obvious work cycle is the Content Management Cycle. This cycle is designed to support the ongoing work of adding, changing, and maintaining content, or application knowledge. There are usually specific persons who are assigned to roles as Content Managers. These persons are generally the subject matter experts who understand the knowledge. Different content management roles may also be defined that require varying levels of skill and knowledge. Different user interfaces may be developed for different roles in order to tailor the level of access and content to each role's specific responsibilities.

There are various types of content management schemes available, depending on the characteristics and use of the content. Some schemes are designed to allow the continuing addition and removal of content records, such as the maintenance of a library of insurance plans. Other schemes are designed around the maintenance of brands. Most are designed around the need to support what we call "change over time".

A typical example requiring content that changes over time is when the content changes are due to underlying regulations and policies. Often this is content that may differ at different known future dates. Phios has developed an approach where the application runs the "current" configuration of content, while the content managers prepare the "next" configuration of the content. At the appropriate time, "next" becomes "current" and the new "next" can begin to be configured again. This flexibility is possible because the Content Management Cycle is kept distinct from the Application Operation Cycle. More importantly, content changes do not require application changes.

2.3.2 Application Operation Cycle

The Application Operation Cycle refers to all of the activities that occur once development is complete. This is sometimes referred to as the lifecycle of an application. Normally this begins with pre-operational testing and deployment, and continues on with post-deployment, support, and maintenance. Application changes begin this cycle over again. Eventually, applications are retired, requiring an orderly transition of an application's knowledge to the replacement application.

When Phios creates a solution, testability and test data are incorporated into the solution. New business solutions need to be brought into production as rapidly as possible, and we work to design applications that maintain a high level of quality with a economical level of testing effort. The more failsafes and automated test solutions, the better, since they shortens cycle times. Phios will often achieve less dependence on testing by building in traps or filters for bad data at the front end, ensuring the integrity of the data downstream. At other times we will use code generation techniques that dramatically reduce errors by reducing the chances for human error.

One frequently overlooked aspect of applications is the need for demonstration platforms. Phios will surface this requirement as early as possible in the development cycle so that creation and maintenance of the demonstration platform is as seamless as possible. Phios often creates an Operator Tool to serve this purpose, providing the ability to set up demonstration accounts and data without disturbing the application's live data. We have found this enormously useful during deployment, but also to demonstrate the solutions to new potential users and customers.

3 The Phios Solution Methodology

Phios uses a classic three-phase process framework for developing and delivering solutions. This approach provides a crisp separation of concerns, allows the right work to be done in the right order, and eliminates the "fog of development". It is a planning intensive approach that minimizes rework and backtracking.

3.1 Pre-construction

The emphasis of the Pre-Construction phase of work is figuring out "what" has to be done and "how" it is to be done. The key activities are domain understanding, selection of a technology approach, and preparation for the rest of the project. During this phase key project participants from both Phios and the customer are identified and assigned roles and responsibilities. Project artifacts needed to guide the project are also produced.

3.1.1 Goal Analysis

In many ways Goal Analysis is the most critical activity of all. During this stage of work the precise meaning of "done" is determined and an agreement is created that allows the business stakeholders to "certify" that the delivered solution meets its stated criteria for completeness and correctness. We want to ensure that the project team understands what they are building, as well as why they are building it, from the customer's perspective.

3.1.2 Domain Analysis

The overall Phios methodology is based on a style of development called Domain Driven Development. This approach is centered on two main ideas.

The first idea is that the language and content of the business domain drives the solution. Domain analysis entails identifying, and making explicit, business domain knowledge in order to properly create and use the solution. Phios does this by using a formal ontological approach to capture the concepts, objects, events, logic, names, and relationships that describe the business domain of the solution. We then use specially designed tools to create what we refer to as the "models" and "configurations" that describe the solution domain.

The second idea is that these models and configurations actually become working parts of the solution. The solution is organized and created in such a way that the integrity of the business domain names and concepts remain explicit and intact.

In fact, the Phios approach is to literally create models and configurations in such a way that they encode what we refer to as the "operational semantics" of the solution. This is a fancy way of saying that the artifacts that describe the solution in business terms become components of the solution. This achieves, what some refer to in somewhat scientific terms, as

a “low friction match” between the business and the technology solution. It also means that the solution is completely comprehensible to the business stakeholders.

Domain analysis encompasses the following areas of specification:

- Reasoning and Logic
- Content and Configurations
- Role, Responsibilities, and Collaborations
- Business Processes and Workflow

All of these specifications are defined independently of the technology that will implement the solution. These specifications serve as a valuable description of the business domain, and also become the basis for the maintainable content knowledge of the solution.

3.1.3 Technical Analysis

In addition to Domain Analysis, a complete technology assessment is done to determine the existing system and data ecology in which the solution will live. This will determine what infrastructure is necessary to connect the new solution to the existing environment.

3.1.4 Project Analysis

The final activity during Pre-Construction is to plan out the solution project. Phios uses a rigorous technique for understanding and describing the construction work to be done. We create a bill of materials for all of the elements of the “whole product solution” that includes separate sections for the application components, content, processes, and infrastructure. Phios produces a pair of documents we refer to as the “Build Manifest” and the “Release Manifest” which are concise visualizations of the project that allow all project stakeholders to envision the components and the sequence of project deliverables.

3.2 Construction

The Construction phase of work is where the actual production-ready software and infrastructure are produced. Driven by the goals and project manifests produced by the Pre-Construction phase, the multi-function team works with regular collaboration with the client to complete the work.

Phios develops solutions using a highly iterative approach. Wherever possible, Phios will sequence the work in such a way that pieces of the solution can be rapidly completed and shown to the customer. This provides ample opportunities to increase feedback and

understanding. As a result of our knowledge-based approach, it is usually possible to review the business logic and content early in the construction phase.

Iteration also helps in the context of moving releases into production. Phios believes in structuring efforts such that as much functionality is released as rapidly as possible, with more releases to follow. Projects that "go dark" for some period of time and spend too long without exposure to the real business environment tend to be less strong than those that release functionality into production on a regular basis. Phios teams operate with this philosophy and are tightly coupled with their customers.

During the construction phase Phios puts particular emphasis on the testing and demonstration of the solution. Most solutions are accompanied by an Operator Tool that supports the creation of test data, and provides a way of operating the solution for both testing and demonstration.

3.3 Post-construction

The Post-Construction phase emphasizes the delivery of the solution to the customer along with the ongoing support of the application, usually by the customer's support staff. Phios solutions will include tools to properly support and address issues, as well as a rigorous process to analyze data coming in from the field and understand the root causes and potential solutions.Phios pays particular attention to the deployment activity when the solution is put into operation. Phios produces appropriate documentation and training materials including materials to support Frequently Asked Questions (FAQs) and troubleshooting. Phios warranties its work and remains available for support, and further development and evolution, of any Phios-supplied solution.

4 Getting Started Working with Phios

The Phios team, with its deep experience, is proven in its ability to help its customers solve their business challenges effectively. Starting to work with Phios involves a detailed conversation about the most important goals and objectives for the client organization, and working with the Phios team to craft a path forward to a solution. We look forward to talking with you about your challenges.